US011087728B1

(12) **United States Patent**
Canberk et al.

(10) **Patent No.: US 11,087,728 B1**
(45) **Date of Patent: Aug. 10, 2021**

(54) **COMPUTER VISION AND MAPPING FOR AUDIO APPLICATIONS**

(71) Applicants: **Ilteris Canberk**, Venice, CA (US); **Donald Giovannini**, Venice, CA (US); **Sana Park**, Santa Monica, CA (US)

(72) Inventors: **Ilteris Canberk**, Venice, CA (US); **Donald Giovannini**, Venice, CA (US); **Sana Park**, Santa Monica, CA (US)

(73) Assignee: **Snap Inc.**, Santa Monica, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **16/724,209**

(22) Filed: **Dec. 21, 2019**

(51) **Int. Cl.**
G10H 1/00 (2006.01)
G06T 7/90 (2017.01)

(52) **U.S. Cl.**
CPC ***G10H 1/0025*** (2013.01); ***G06T 7/90*** (2017.01); ***G10H 1/0041*** (2013.01); *G06T 2207/10024* (2013.01); *G06T 2207/30204* (2013.01); *G10H 2220/101* (2013.01); *G10H 2220/445* (2013.01); *G10H 2220/455* (2013.01)

(58) **Field of Classification Search**
CPC G10H 1/0025; G10H 1/0041; G10H 2220/101; G10H 2220/445; G10H 2220/455; G06T 7/90; G06T 2207/10024; G06T 2207/30204
USPC 84/609
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,353,378 A * 10/1994 Hoffman G02C 11/10 704/258
5,543,816 A * 8/1996 Heacock G02B 13/18 345/8
5,616,030 A * 4/1997 Watson G09B 9/08 434/29
5,924,868 A * 7/1999 Rod F41G 3/2611 348/158
6,963,656 B1 * 11/2005 Persaud G09B 21/006 348/62
7,283,983 B2 * 10/2007 Dooley G06F 3/017 706/20
10,013,961 B1 * 7/2018 Campbell G10G 1/02
2011/0254832 A1 * 10/2011 Wilson G10H 1/0025 345/418
2012/0306914 A1 * 12/2012 Sepulveda G06T 15/50 345/629

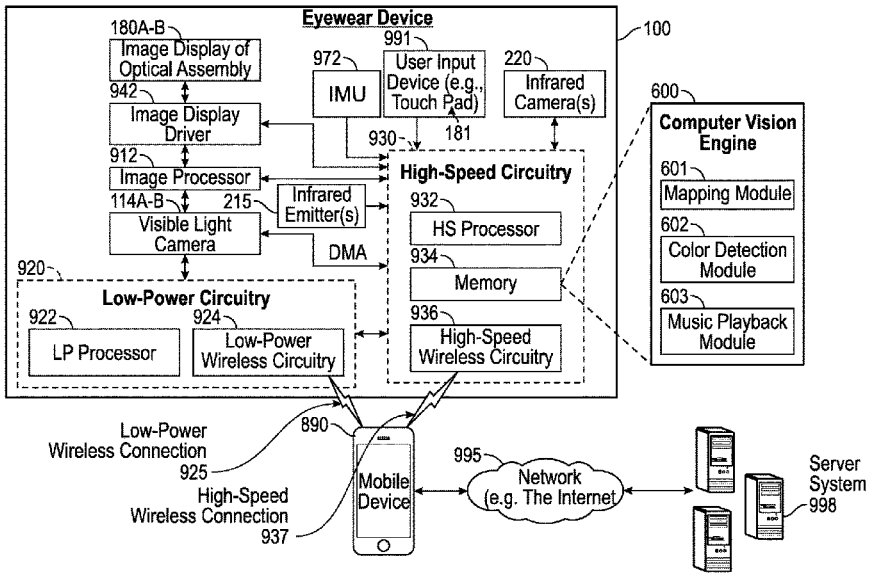
(Continued)

Primary Examiner — Christina M Schreiber
(74) *Attorney, Agent, or Firm* — Culhane Meadows PLLC; Stephen J. Weed

(57) **ABSTRACT**

Systems, devices, media, and methods are presented for playing audio sounds, such as music, on a portable electronic device using a digital color image of a note matrix on a map. A computer vision engine, in an example implementation, includes a mapping module, a color detection module, and a music playback module. The camera captures a color image of the map, including a marker and a note matrix. Based on the color image, the computer vision engine detects a token color value associated with each field. Each token color value is associated with a sound sample from a specific musical instrument. A global state map is stored in memory, including the token color value and location of each field in the note matrix. The music playback module, for each column, in order, plays the notes associated with one or more the rows, using the corresponding sound sample, according to the global state map.

22 Claims, 11 Drawing Sheets



(56) **References Cited**

U.S. PATENT DOCUMENTS

2013/0127980	A1*	5/2013	Haddick	G02B 27/0093 348/14.08
2013/0278631	A1*	10/2013	Border	G06F 3/04842 345/633
2014/0248950	A1*	9/2014	Tosas Bautista	G06F 1/1694 463/31
2017/0091983	A1*	3/2017	Sebastian	G06T 15/205
2018/0107278	A1*	4/2018	Goel	G06F 3/162
2018/0139565	A1*	5/2018	Norris	H04S 1/007
2018/0210064	A1*	7/2018	Send	G01C 3/08
2018/0218541	A1*	8/2018	Brown, III	H04S 7/303
2018/0357991	A1*	12/2018	Barry	G10H 1/383
2019/0045125	A1*	2/2019	Jarvenpaa	H04N 5/23267

* cited by examiner

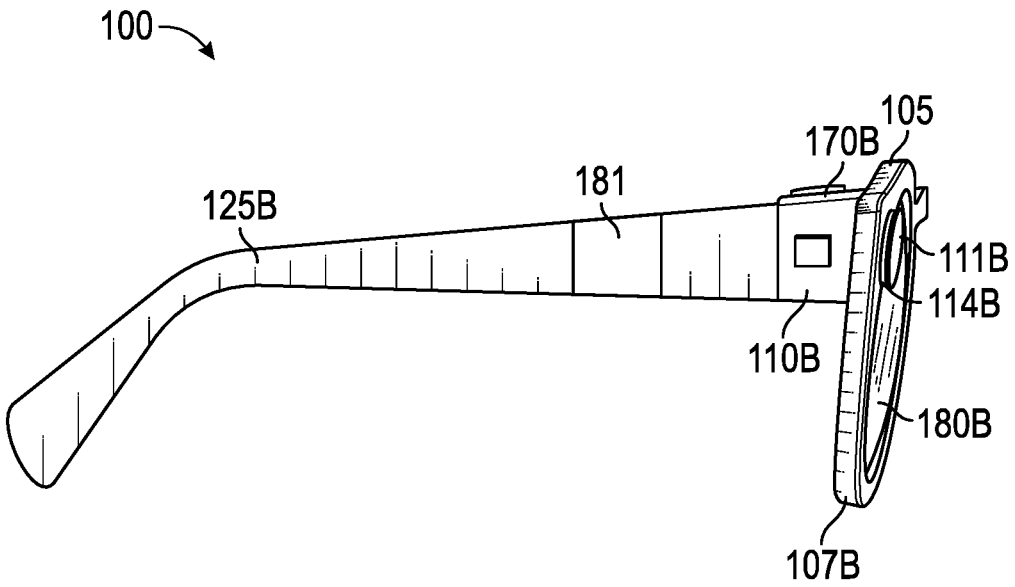


FIG. 1A

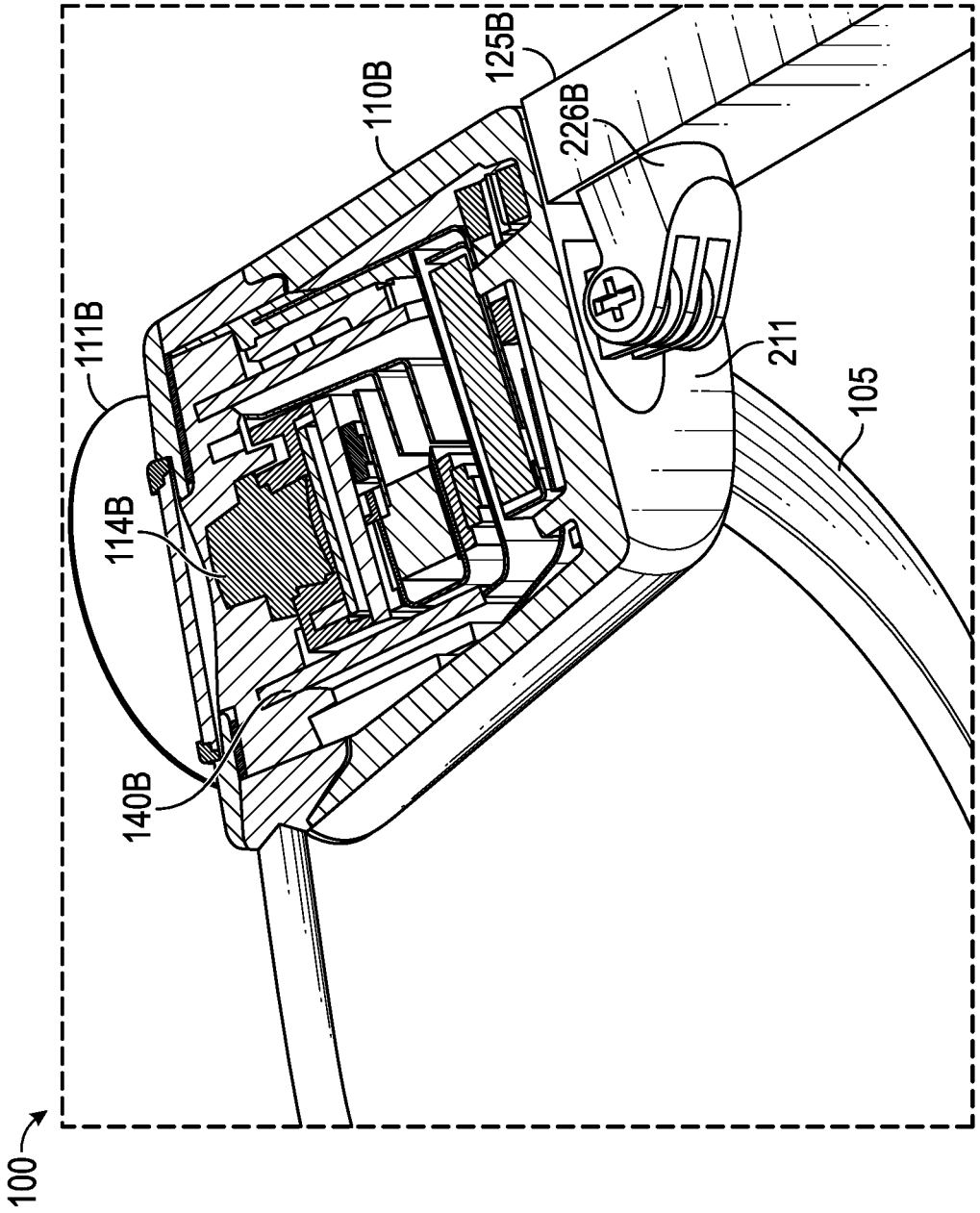


FIG. 1B

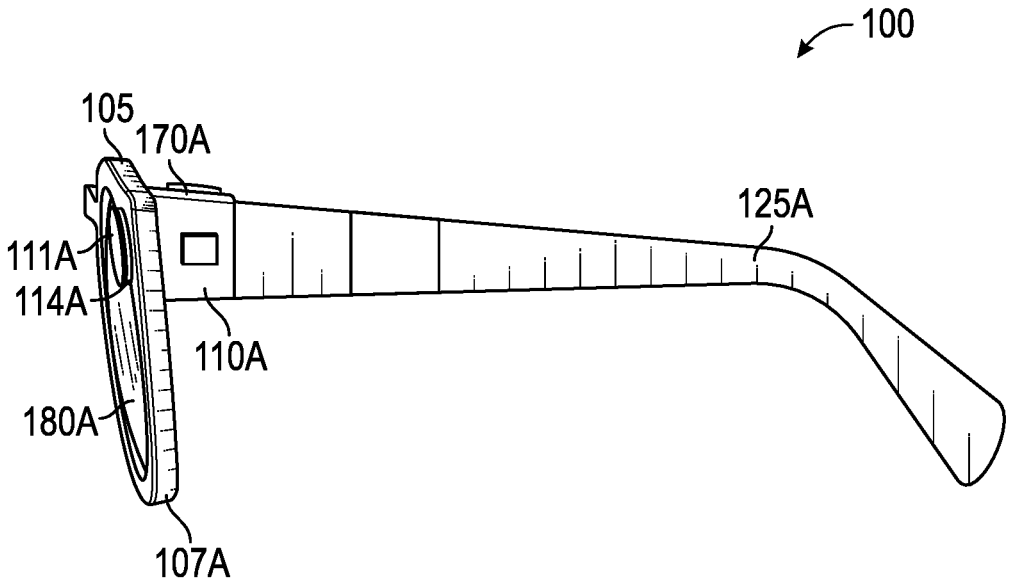


FIG. 1C

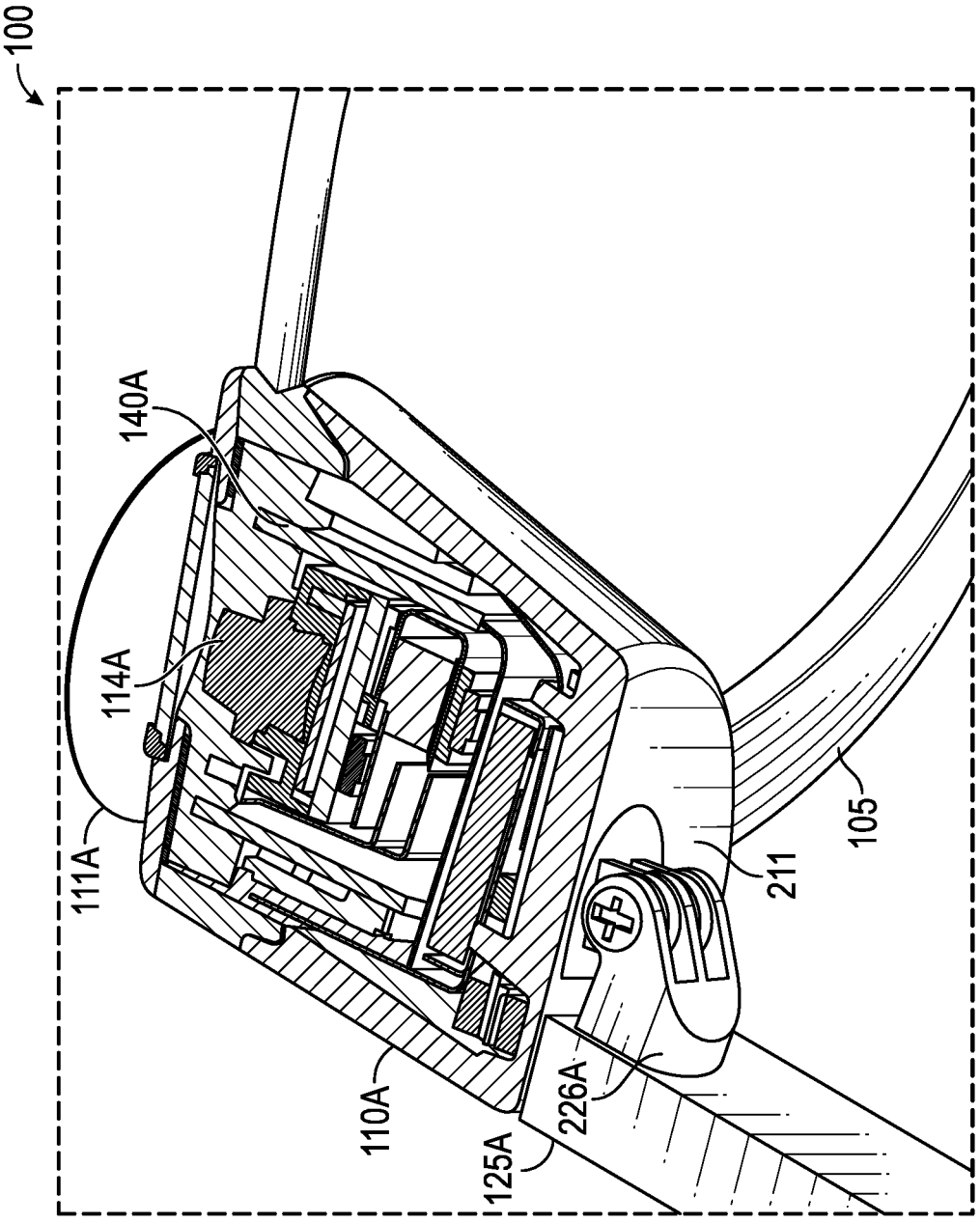


FIG. 1D

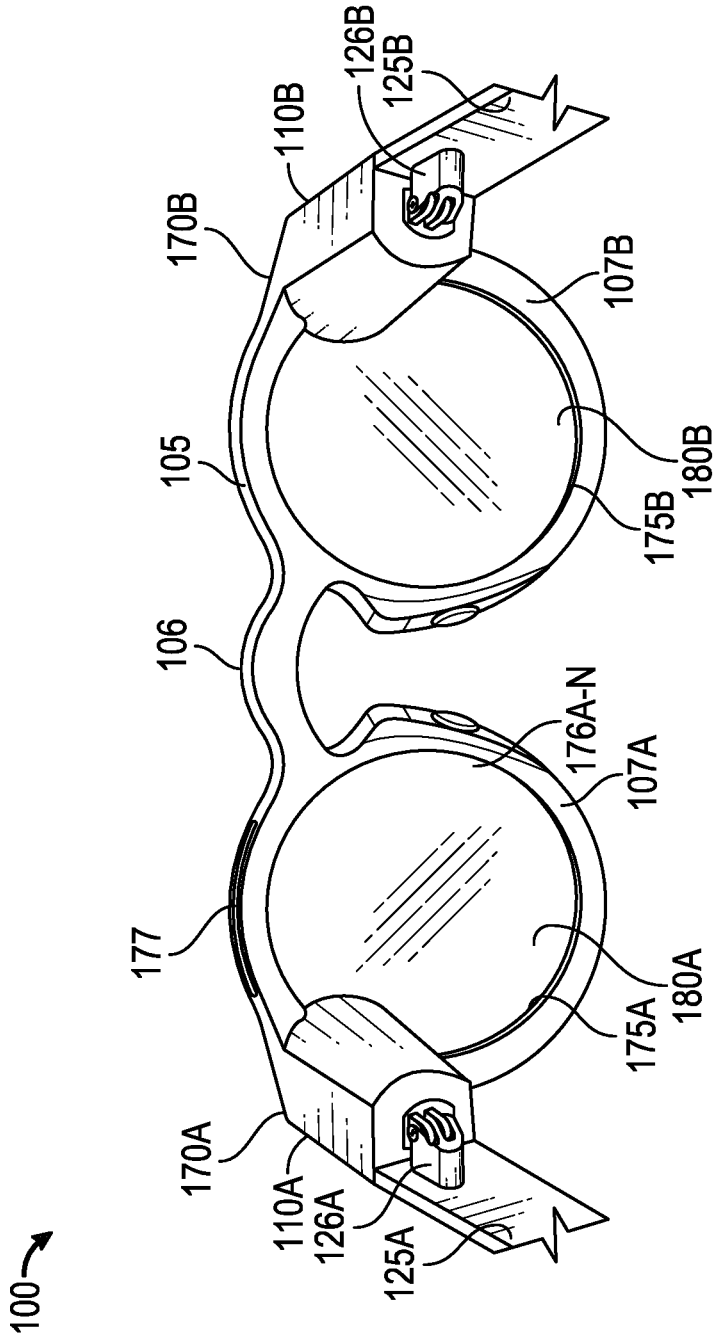

100
110A
126A
125A
170A
177
176A-N
107A
180A
175A
105
106
110B
126B
125B
170B
107B
180B
175B

FIG. 2A

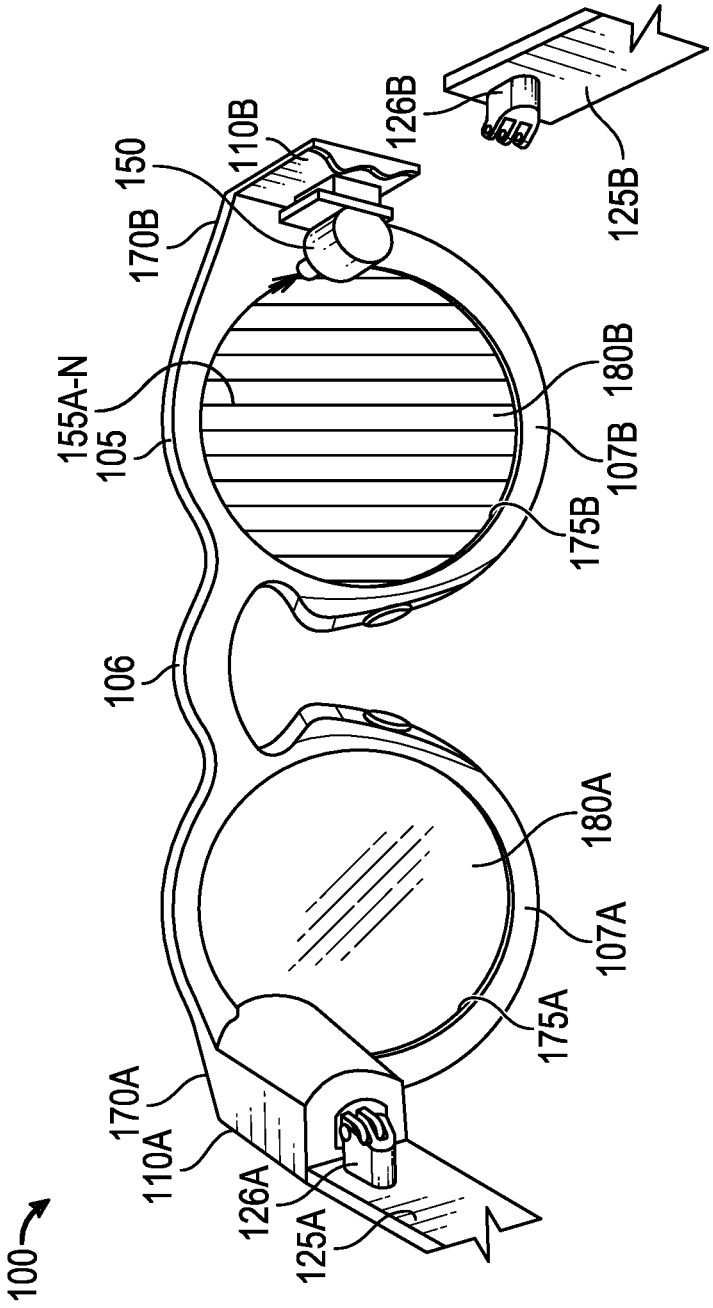

100
110A
170A
126A
125A
175A
107A
180A
106
105
155A-N
170B
150
110B
126B
125B
175B
107B
180B

FIG. 2B

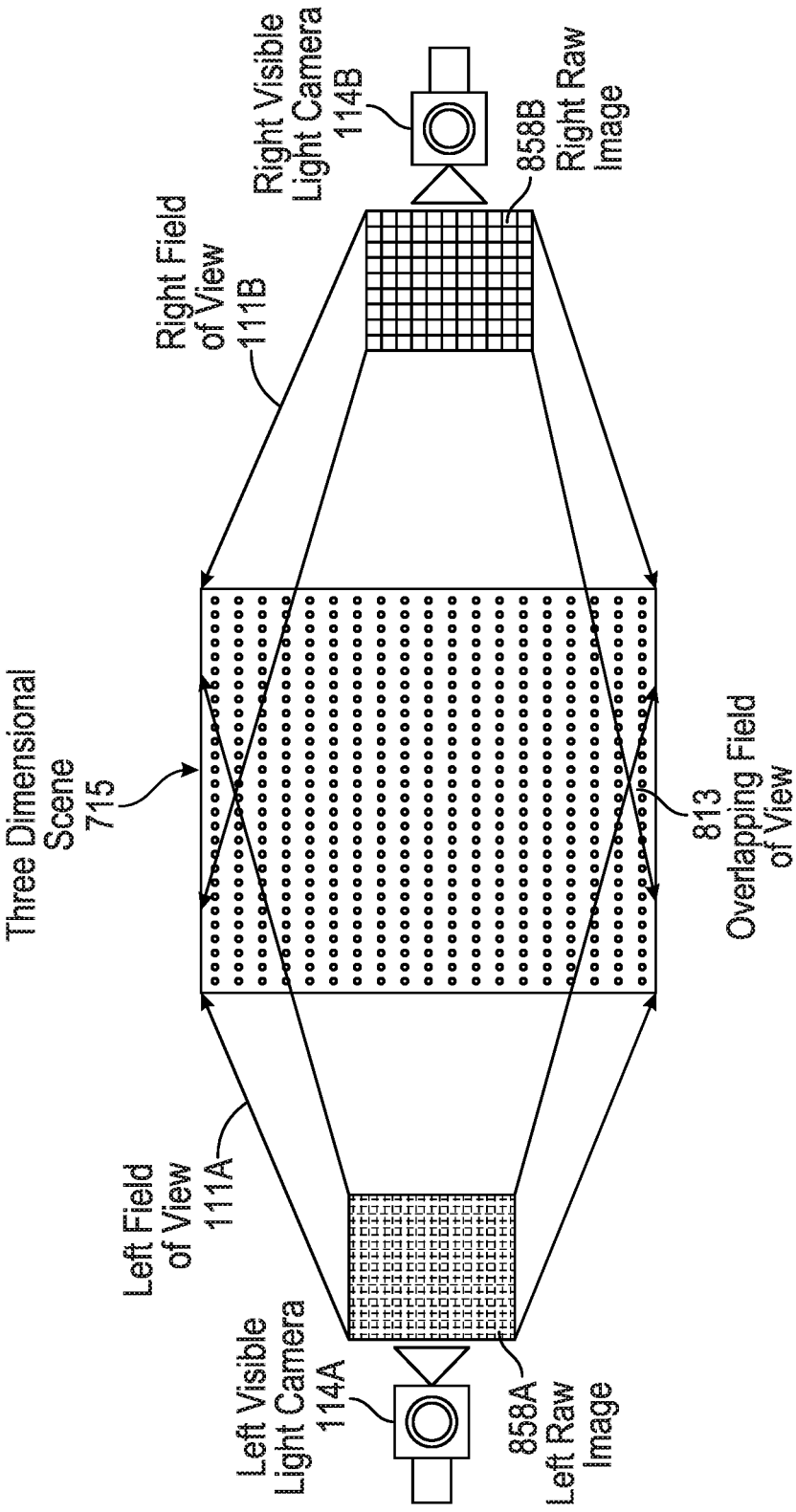

Right Visible Light Camera 114B
858B Right Raw Image
Right Field of View 111B
Three Dimensional Scene 715
813 Overlapping Field of View
Left Field of View 111A
Left Visible Light Camera 114A
858A Left Raw Image

FIG. 3

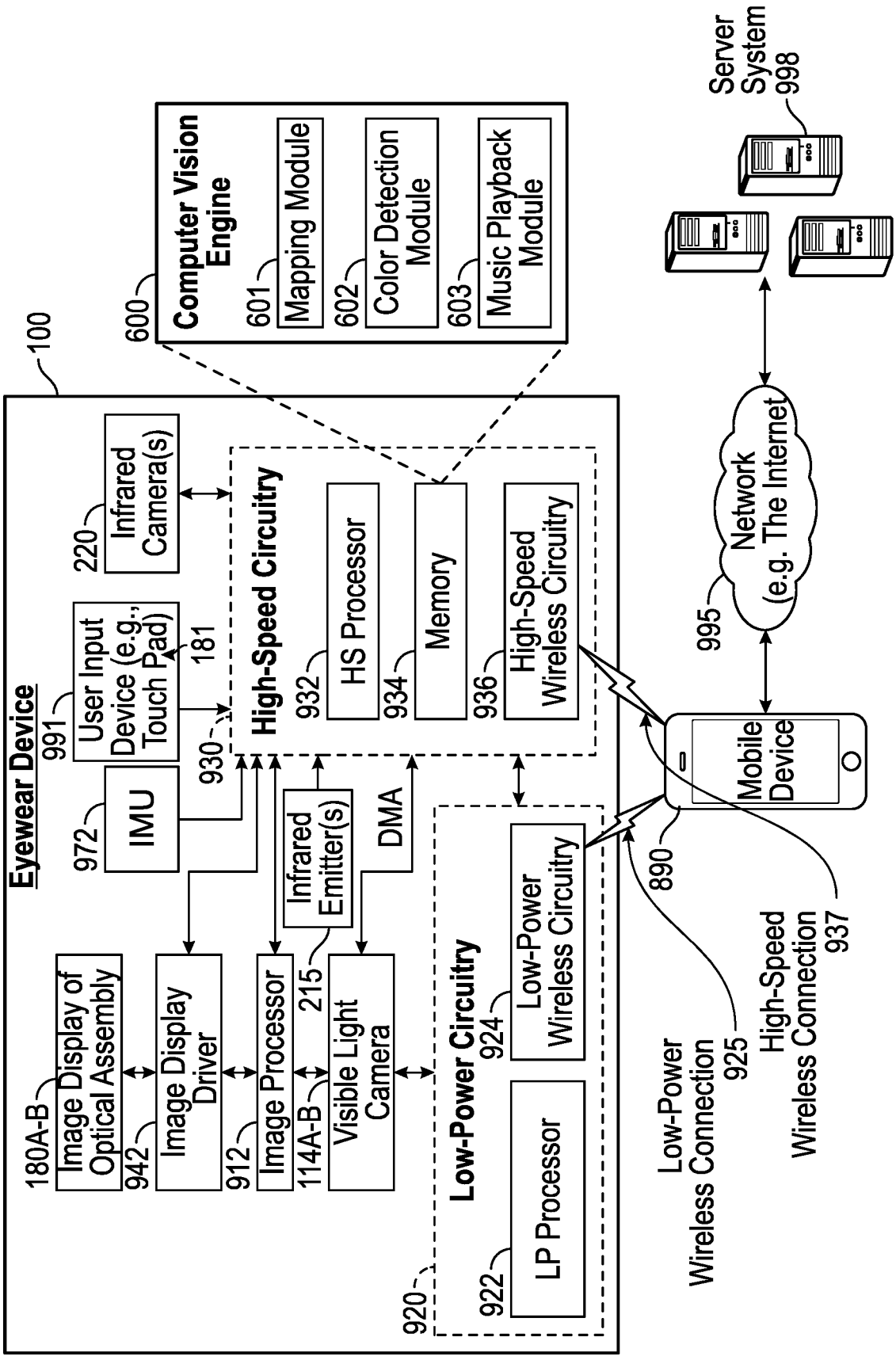

Server System 998
Computer Vision Engine
600
601 Mapping Module
602 Color Detection Module
603 Music Playback Module
100
Eyewear Device
220 Infrared Camera(s)
991 User Input Device (e.g., Touch Pad)
181
930
High-Speed Circuitry
932 HS Processor
934 Memory
936 High-Speed Wireless Circuitry
972 IMU
Infrared Emitter(s) 215
DMA
180A-B Image Display of Optical Assembly
942 Image Display Driver
912 Image Processor
114A-B Visible Light Camera
920
Low-Power Circuitry
924 Low-Power Wireless Circuitry
922 LP Processor
995 Network (e.g. The Internet)
Mobile Device
890
Low-Power Wireless Connection 925
High-Speed Wireless Connection 937

FIG. 4

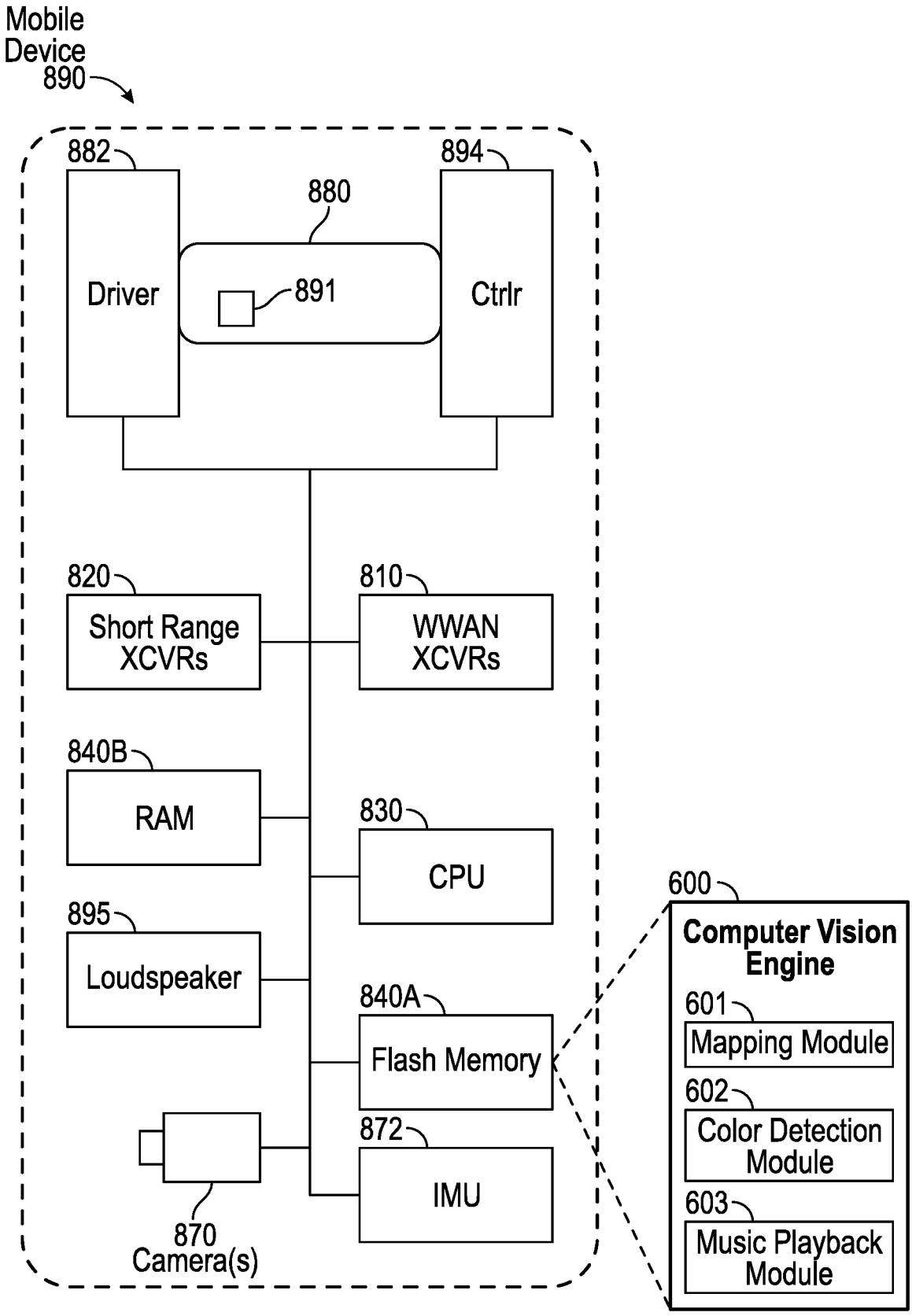

Mobile Device
890
882
894
880
Driver
891
Ctrlr
820
Short Range XCVRs
810
WWAN XCVRs
840B
RAM
830
CPU
895
Loudspeaker
840A
Flash Memory
872
IMU
870
Camera(s)
600
Computer Vision Engine
601
Mapping Module
602
Color Detection Module
603
Music Playback Module

FIG. 5

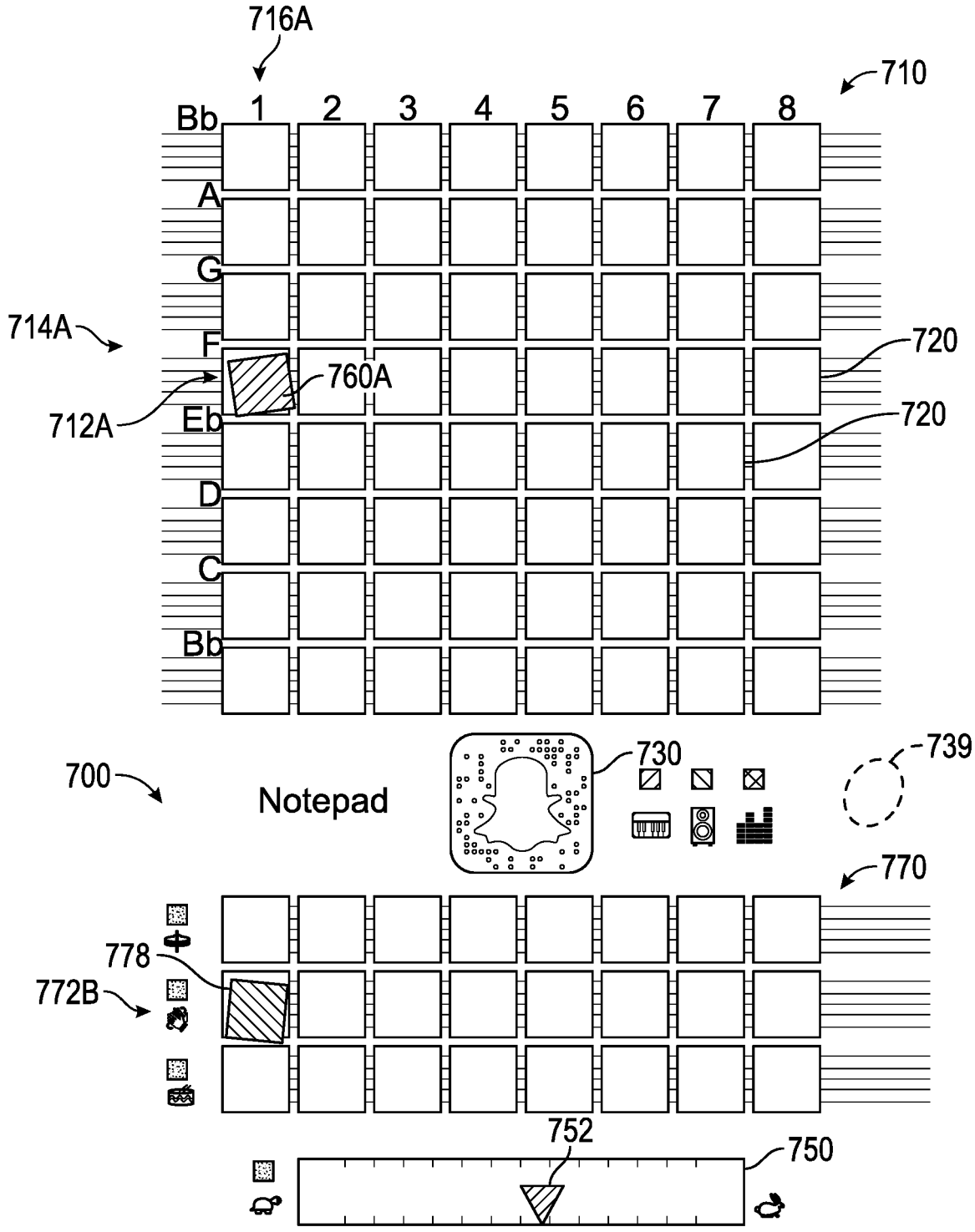

716A
710
1
2
3
4
5
6
7
8
Bb
A
G
F
Eb
D
C
Bb
714A
712A
760A
720
720
700
Notepad
730
739
770
778
772B
752
750

FIG. 6

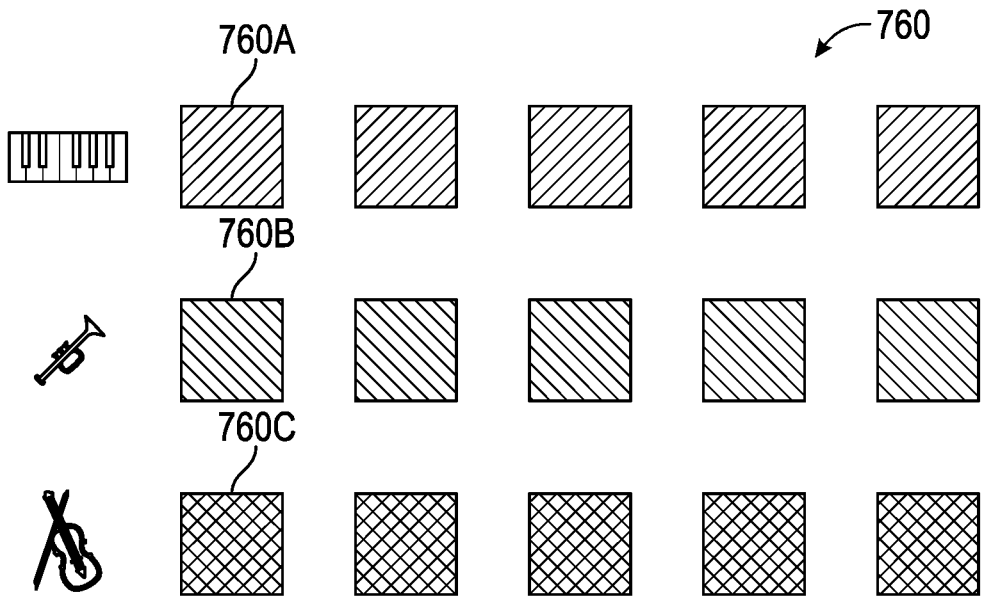


FIG. 7

COMPUTER VISION AND MAPPING FOR AUDIO APPLICATIONS

TECHNICAL FIELD

Examples set forth in the present disclosure relate to portable electronic devices, including wearable devices such as eyewear. More particularly, but not by way of limitation, the present disclosure describes computer vision and mapping systems and methods for audio applications, such as playing musical sounds.

BACKGROUND

Many types of computers and electronic devices available today, including mobile devices (e.g., smartphones, tablets, and laptops), and wearable devices (e.g., smartglasses, digital eyewear, headwear, headgear, and head-mounted displays), include digital cameras, display screens, and interfaces through which a user can interact with displayed content.

BRIEF DESCRIPTION OF THE DRAWINGS

Features of the various implementations disclosed will be readily understood from the following detailed description, in which reference is made to the appending drawing figures. A reference numeral is used with each element in the description and throughout the several views of the drawing. When a plurality of similar elements is present, a single reference numeral may be assigned to like elements, with an added lower-case letter referring to a specific element.

The various elements shown in the figures are not drawn to scale unless otherwise indicated. The dimensions of the various elements may be enlarged or reduced in the interest of clarity. The several figures depict one or more implementations and are presented by way of example only and should not be construed as limiting. Included in the drawing are the following figures:

FIG. **1**A is a side view (right) of an example hardware configuration of an eyewear device with a primary input surface, which may be utilized in a selective control and transition system;

FIG. **1**B is a top, partly sectional view of a right chunk of the eyewear device of FIG. **1**A depicting a right visible-light camera, and a circuit board;

FIG. **1**C is a side view (left) of an example hardware configuration of the eyewear device of FIG. **1**A, which shows a left visible-light camera;

FIG. **1**D is a top, partly sectional view of a left chunk of the eyewear device of FIG. **1**C depicting the left visible-light camera, and a circuit board;

FIGS. **2**A and **2**B are rear views of example hardware configurations of an eyewear device utilized in the selective control and transition system;

FIG. **3** is a diagrammatic depiction of a three-dimensional scene, a left raw image captured by a left visible-light camera, and a right raw image captured by a right visible-light camera;

FIG. **4** is a functional block diagram of an example system including a mobile device, a wearable device (e.g., an eyewear device), a computer vision engine, and a server system connected via various networks;

FIG. **5** is a diagrammatic representation of an example hardware configuration for a mobile device including the computer vision engine of FIG. **4**;

FIG. **6** is an illustration of an example map, including a marker, a note matrix, a percussion matrix, and a tempo scale, suitable for use with the computer vision engine of FIG. **4**; and

FIG. **7** is an illustration of a set of example tokens ready to be cut and placed on the example map of FIG. **6**.

DETAILED DESCRIPTION

Various implementations and details are described with reference to an example: a computer vision engine for playing a musical composition on a portable electronic device using a digital color image of a note matrix on a map. A mobile device, such as a smartphone, includes a processor, a memory, a camera, a loudspeaker, and a display screen. The computer vision engine, in an example implementation, includes a mapping module, a color detection module, and a music playback module. The camera captures a color image of the map, including a marker and a note matrix (in rows and columns, similar to standard musical notation, with a note in each field). From the color image, the computer vision engine detects a token color value associated with each field. Each token color value is associated with a sound sample from a specific musical instrument. A global state map is stored in memory, including the token color value and location of each field in the note matrix. The music playback module, for each column, in order, plays the notes associated with one or more the rows, using the corresponding sound sample, according to the global state map.

The following detailed description includes systems, methods, techniques, instruction sequences, and computing machine program products illustrative of examples set forth in the disclosure. Numerous details and examples are included for the purpose of providing a thorough understanding of the disclosed subject matter and its relevant teachings. Those skilled in the relevant art, however, may understand how to apply the relevant teachings without such details. Aspects of the disclosed subject matter are not limited to the specific devices, systems, and method described because the relevant teachings can be applied or practice in a variety of ways. The terminology and nomenclature used herein is for the purpose of describing particular aspects only and is not intended to be limiting. In general, well-known instruction instances, protocols, structures, and techniques are not necessarily shown in detail.

The term "coupled" or "connected" as used herein refers to any logical, optical, physical, or electrical connection, including a link or the like by which the electrical or magnetic signals produced or supplied by one system element are imparted to another coupled or connected system element. Unless described otherwise, coupled or connected elements or devices are not necessarily directly connected to one another and may be separated by intermediate components, elements, or communication media, one or more of which may modify, manipulate, or carry the electrical signals. The term "on" means directly supported by an element or indirectly supported by the element through another element integrated into or supported by the element.

The orientations of the eyewear device, the handheld device, associated components and any other complete devices incorporating a camera and/or an inertial measurement unit such as shown in any of the drawings, are given by way of example only, for illustration and discussion purposes. In operation, the eyewear device may be oriented in any other direction suitable to the particular application of the eyewear device; for example, up, down, sideways, or any other orientation. Also, to the extent used herein, any direc-

tional term, such as front, rear, inward, outward, toward, left, right, lateral, longitudinal, up, down, upper, lower, top, bottom, side, horizontal, vertical, and diagonal are used by way of example only, and are not limiting as to the direction or orientation of any camera and/or inertial measurement unit as constructed as otherwise described herein.

Additional objects, advantages and novel features of the examples will be set forth in part in the following description, and in part will become apparent to those skilled in the art upon examination of the following and the accompanying drawings or may be learned by production or operation of the examples. The objects and advantages of the present subject matter may be realized and attained by means of the methodologies, instrumentalities and combinations particularly pointed out in the appended claims.

Reference now is made in detail to the examples illustrated in the accompanying drawings and discussed below.

FIG. **1**A is a side view (right) of an example hardware configuration of an eyewear device **100** which includes a touch-sensitive input device or touchpad **181**. As shown, the touchpad **181** may have a boundary that is subtle and not easily seen; alternatively, the boundary may be plainly visible and/or include a raised or otherwise tactile edge that provides feedback to the user about the location and boundary of the touchpad **181**. In other implementations, the eyewear **100** may include a touchpad on the left side.

The surface of the touchpad **181** is configured to detect finger touches, taps, and gestures (e.g., moving touches) for use with a GUI displayed by the eyewear, on a display, to allow the user to navigate through and select menu options in an intuitive manner, which enhances and simplifies the user experience.

Detection of finger inputs on the touchpad **181** can enable several functions. For example, touching anywhere on the touchpad **181** may cause the GUI to display and/or highlight an item on the screen of the image display, which may be projected onto at least one of the optical assemblies **180**A, **180**B. Double tapping on the touchpad **181** may select an item or icon. Sliding or swiping a finger in a particular direction (e.g., from front to back, back to front, up to down, or down to) may cause the items or icons to slide or scroll in a particular direction; for example, to move to a next item, icon, video, image, page, or slide. Sliding the finger in another direction may slide or scroll in the opposite direction; for example, to move to a previous item, icon, video, image, page, or slide. The touchpad **181** can be virtually anywhere on the eyewear device **100**.

In one example, when the identified finger gesture is a single tap on the touchpad **181**, this initiates selection or pressing of a graphical user interface element in the image presented on the image display of the optical assembly **180**A, **180**B. An adjustment to the image presented on the image display of the optical assembly **180**A, **180**B based on the identified finger gesture can be a primary action which selects or submits the graphical user interface element on the image display of the optical assembly **180**A, **180**B for further display or execution.

As shown, the eyewear **100** includes a right visible-light camera **114**B. As further described herein, two cameras **114**A, **114**B capture image information for a scene from two separate viewpoints. The two captured images may be used to project a three-dimensional display onto a screen for viewing with 3D glasses.

The eyewear device **100** includes a right optical assembly **180**B with an image display to present images, such as depth images. As shown in FIGS. **1**A and **1**B, the eyewear device **100** includes the right visible-light camera **114**B. The eyewear device **100** can include multiple visible-light cameras **114**A, **114**B that form a passive type of three-dimensional camera, such as stereo camera, of which the right visible-light camera **114**B is located on a right chunk **110**B. As shown in FIGS. **1**C-D, the eyewear device **100** also includes a left visible-light camera **114**A.

Left and right visible-light cameras **114**A, **114**B are sensitive to the visible-light range wavelength. Each of the visible-light cameras **114**A, **114**B have a different frontward facing field of view which are overlapping to enable generation of three-dimensional depth images, for example, right visible-light camera **114**B depicts a right field of view **111**B. Generally, a "field of view" is the part of the scene that is visible through the camera at a particular position and orientation in space. The fields of view **111**A and **111**B have an overlapping field of view **813**. Objects or object features outside the field of view **111**A, **111**B when the visible-light camera captures the image are not recorded in a raw image (e.g., photograph or picture). The field of view describes an angle range or extent, which the image sensor of the visible-light camera **114**A, **114**B picks up electromagnetic radiation of a given scene in a captured image of the given scene. Field of view can be expressed as the angular size of the view cone, i.e., an angle of view. The angle of view can be measured horizontally, vertically, or diagonally.

In an example, visible-light cameras **114**A, **114**B have a field of view with an angle of view between 15° to 30°, for example 24°, and have a resolution of 480×480 pixels. The "angle of coverage" describes the angle range that a lens of visible-light cameras **114**A, **114**B or infrared camera **220** (see FIG. **2**A) can effectively image. Typically, the camera lens produces an image circle that is large enough to cover the film or sensor of the camera completely, possibly including some vignetting toward the edge. If the angle of coverage of the camera lens does not fill the sensor, the image circle will be visible, typically with strong vignetting toward the edge, and the effective angle of view will be limited to the angle of coverage.

Examples of such visible-light cameras **114**A, **114**B include a high-resolution complementary metal-oxide-semiconductor (CMOS) image sensor and a digital VGA camera (video graphics array) capable of resolutions of **640***p* (e.g., 640×480 pixels for a total of 0.3 megapixels), 720p, or 1080p. Other examples of visible-light cameras **114**A, **114**B that can capture high-definition (HD) still images and store them at a resolution of 1642 by 1642 pixels (or greater); and/or record high-definition video at a high frame rate (e.g., thirty to sixty frames per second or more) and store the recording at a resolution of 1216 by 1216 pixels (or greater).

The eyewear device **100** may capture image sensor data from the visible-light cameras **114**A, **114**B along with geolocation data, digitized by an image processor, for storage in a memory. The left and right raw images captured by respective visible-light cameras **114**A, **114**B are in the two-dimensional space domain and comprise a matrix of pixels on a two-dimensional coordinate system that includes an X-axis for horizontal position and a Y-axis for vertical position. Each pixel includes a color attribute value (e.g., a red pixel light value, a green pixel light value, and/or a blue pixel light value); and a position attribute (e.g., an X-axis coordinate and a Y-axis coordinate).

In order to capture stereo images for later display as a three-dimensional projection, the image processor **912** (shown in FIG. **4**) may be coupled to the visible-light cameras **114**A, **114**B to receive and store the visual image information. A timestamp for each image may be added by the image processor **912** or another processor which controls

operation of the visible-light cameras **114**A, **114**B, which act as a stereo camera to simulate human binocular vision. The timestamp on each pair of images allows the images to be displayed together as part of a three-dimensional projection. Three-dimensional projections create an immersive, life-like experience that is desirable in a variety of contexts, including virtual reality (VR) and video gaming.

FIG. **3** is a diagrammatic depiction of a three-dimensional scene **715**, a left raw image **858**A captured by a left visible-light camera **114**A, and a right raw image **858**B captured by a right visible-light camera **114**B. The left field of view **111**A may overlap, as shown, with the right field of view **111**B. The overlapping field of view **813** represents that portion of the image captured by both cameras **114**A, **114**B. The term 'overlapping' when referring to field of view means the matrix of pixels in the generated raw images overlap by thirty percent (30%) or more. 'Substantially overlapping' means the matrix of pixels in the generated raw images—or in the infrared image of scene—overlap by fifty percent (50%) or more. As described herein, the two raw images **858**A, **858**B may be processed to include a timestamp, which allows the images to be displayed together as part of a three-dimensional projection.

For the capture of stereo images, as illustrated in FIG. **3**, a pair of raw red, green, and blue (RGB) images are captured of a real scene **715** at a given moment in time—a left raw image **858**A captured by the left camera **114**A and right raw image **858**B captured by the right camera **114**B. When the pair of raw images **858**A, **858**B are processed (e.g., by the image processor **912**), depth images are generated. The generated depth images may be viewed on an optical assembly **180**A, **180**B of an eyewear device, on another display (e.g., the image display **880** on a mobile device **890**), or on a screen.

The generated depth images are in the three-dimensional space domain and can comprise a matrix of vertices on a three-dimensional location coordinate system that includes an X axis for horizontal position (e.g., length), a Y axis for vertical position (e.g., height), and a Z axis for depth (e.g., distance). Each vertex may include a color attribute (e.g., a red pixel light value, a green pixel light value, and/or a blue pixel light value); a position attribute (e.g., an X location coordinate, a Y location coordinate, and a Z location coordinate); a texture attribute and/or a reflectance attribute. The texture attribute quantifies the perceived texture of the depth image, such as the spatial arrangement of color or intensities in a region of vertices of the depth image.

In one example, the system described herein includes the eyewear device **100**, which includes a frame **105** and a left temple **125**A extending from a left lateral side **170**A of the frame **105** and a right temple **125**B extending from a right lateral side **170**B of the frame **105**. The eyewear device **100** may further include at least two visible-light cameras **114**A, **114**B which may have overlapping fields of view. In one example, the eyewear device **100** includes a left visible-light camera **114**A with a left field of view **111**A, as illustrated in FIG. **3**. The left camera **114**A is connected to the frame **105** or the left temple **125**A to capture a left raw image **858**A from the left side of scene **715**. The eyewear device **100** further includes a right visible-light camera **114**B with a right field of view **111**B. The right camera **114**B is connected to the frame **105** or the right temple **125**B to capture a right raw image **858**B from the right side of scene **715**.

FIG. **1**B is a top cross-sectional view of a right chunk **110**B of the eyewear device **100** of FIG. **1**A depicting the right visible-light camera **114**B of the camera system, and a circuit board. FIG. **1**C is a side view (left) of an example hardware configuration of an eyewear device **100** of FIG. **1**A, which shows a left visible-light camera **114**A of the camera system. FIG. **1**D is a top cross-sectional view of a left chunk **110**A of the eyewear device of FIG. **1**C depicting the left visible-light camera **114**A of the three-dimensional camera, and a circuit board. Construction and placement of the left visible-light camera **114**A is substantially similar to the right visible-light camera **114**B, except the connections and coupling are on the left lateral side **170**A. As shown in the example of FIG. **1**B, the eyewear device **100** includes the right visible-light camera **114**B and a circuit board **140**B, which may be a flexible printed circuit board (PCB). The right hinge **226**B connects the right chunk **110**B to a right temple **125**B of the eyewear device **100**. In some examples, components of the right visible-light camera **114**B, the flexible PCB **140**B, or other electrical connectors or contacts may be located on the right temple **125**B or the right hinge **226**B.

The right chunk **110**B includes chunk body **211** and a chunk cap, with the chunk cap omitted in the cross-section of FIG. **1**B. Disposed inside the right chunk **110**B are various interconnected circuit boards, such as PCBs or flexible PCBs, that include controller circuits for right visible-light camera **114**B, microphone(s), low-power wireless circuitry (e.g., for wireless short-range network communication via Bluetooth™), high-speed wireless circuitry (e.g., for wireless local area network communication via WiFi).

The right visible-light camera **114**B is coupled to or disposed on the flexible PCB **140**B and covered by a visible-light camera cover lens, which is aimed through opening(s) formed in the frame **105**. For example, the right rim **107**B of the frame **105**, shown in FIG. **2**A, is connected to the right chunk **110**B and includes the opening(s) for the visible-light camera cover lens. The frame **105** includes a front side configured to face outward and away from the eye of the user. The opening for the visible-light camera cover lens is formed on and through the front or outward-facing side of the frame **105**. In the example, the right visible-light camera **114**B has an outward-facing field of view **111**B (shown in FIG. **3**) with a line of sight or perspective that is correlated with the right eye of the user of the eyewear device **100**. The visible-light camera cover lens can also be adhered to a front side or outward-facing surface of the right chunk **110**B in which an opening is formed with an outward-facing angle of coverage, but in a different outwardly direction. The coupling can also be indirect via intervening components.

As shown in FIG. **1**B, flexible PCB **140**B is disposed inside the right chunk **110**B and is coupled to one or more other components housed in the right chunk **110**B. Although shown as being formed on the circuit boards of the right chunk **110**B, the right visible-light camera **114**B can be formed on the circuit boards of the left chunk **110**A, the temples **125**A, **125**B, or the frame **105**.

FIGS. **2**A and **2**B are perspective views, from the rear, of example hardware configurations of the eyewear device **100**, including two different types of image displays. The eyewear device **100** is sized and shaped in a form configured for wearing by a user; the form of eyeglasses is shown in the example. The eyewear device **100** can take other forms and may incorporate other types of frameworks; for example, a headgear, a headset, or a helmet.

In the eyeglasses example, eyewear device **100** includes a frame **105** including a left rim **107**A connected to a right rim **107**B via a bridge **106** adapted to be supported by a nose of the user. The left and right rims **107**A, **107**B include

respective apertures **175**A, **175**B, which hold a respective optical element **180**A, **180**B, such as a lens and a display device. As used herein, the term "lens" is meant to include transparent or translucent pieces of glass or plastic having curved and/or flat surfaces that cause light to converge/diverge or that cause little or no convergence or divergence.

Although shown as having two optical elements **180**A, **180**B, the eyewear device **100** can include other arrangements, such as a single optical element (or it may not include any optical element **180**A, **180**B), depending on the application or the intended user of the eyewear device **100**. As further shown, eyewear device **100** includes a left chunk **110**A adjacent the left lateral side **170**A of the frame **105** and a right chunk **110**B adjacent the right lateral side **170**B of the frame **105**. The chunks **110**A, **110**B may be integrated into the frame **105** on the respective sides **170**A, **170**B (as illustrated) or implemented as separate components attached to the frame **105** on the respective sides **170**A, **170**B. Alternatively, the chunks **110**A, **110**B may be integrated into temples (not shown) attached to the frame **105**.

In one example, the image display of optical assembly **180**A, **180**B includes an integrated image display. As shown in FIG. **2**A, each optical assembly **180**A, **180**B includes a suitable display matrix **177**, such as a liquid crystal display (LCD), an organic light-emitting diode (OLED) display, or any other such display. Each optical assembly **180**A, **180**B also includes an optical layer or layers **176**, which can include lenses, optical coatings, prisms, mirrors, waveguides, optical strips, and other optical components in any combination. The optical layers **176**A, **176**B, . . . **176**N (shown as **176**A-N in FIG. **2**A and herein) can include a prism having a suitable size and configuration and including a first surface for receiving light from a display matrix and a second surface for emitting light to the eye of the user. The prism of the optical layers **176**A-N extends over all or at least a portion of the respective apertures **175**A, **175**B formed in the left and right rims **107**A, **107**B to permit the user to see the second surface of the prism when the eye of the user is viewing through the corresponding left and right rims **107**A, **107**B. The first surface of the prism of the optical layers **176**A-N faces upwardly from the frame **105** and the display matrix **177** overlies the prism so that photons and light emitted by the display matrix **177** impinge the first surface. The prism is sized and shaped so that the light is refracted within the prism and is directed toward the eye of the user by the second surface of the prism of the optical layers **176**A-N. In this regard, the second surface of the prism of the optical layers **176**A-N can be convex to direct the light toward the center of the eye. The prism can optionally be sized and shaped to magnify the image projected by the display matrix **177**, and the light travels through the prism so that the image viewed from the second surface is larger in one or more dimensions than the image emitted from the display matrix **177**.

In one example, the optical layers **176**A-N may include an LCD layer that is transparent (keeping the lens open) unless and until a voltage is applied which makes the layer opaque (closing or blocking the lens). The image processor **912** on the eyewear device **100** may execute programming to apply the voltage to the LCD layer in order to create an active shutter system, making the eyewear device **100** suitable for viewing visual content when displayed as a three-dimensional projection. Technologies other than LCD may be used for the active shutter mode, including other types of reactive layers that are responsive to a voltage or another type of input.

In another example, the image display device of optical assembly **180**A, **180**B includes a projection image display as shown in FIG. **2**B. Each optical assembly **180**A, **180**B includes a laser projector **150**, which is a three-color laser projector using a scanning mirror or galvanometer. During operation, an optical source such as a laser projector **150** is disposed in or on one of the temples **125**A, **125**B of the eyewear device **100**. Optical assembly **180**B in this example includes one or more optical strips **155**A, **155**B, . . . **155**N (shown as **155**A-N in FIG. **2**B) which are spaced apart and across the width of the lens of each optical assembly **180**A, **180**B and/or across a depth of the lens between the front surface and the rear surface of the lens.

As the photons projected by the laser projector **150** travel across the lens of each optical assembly **180**A, **180**B, the photons encounter the optical strips **155**A-N. When a particular photon encounters a particular optical strip, the photon is either redirected toward the user's eye, or it passes to the next optical strip. A combination of modulation of laser projector **150**, and modulation of optical strips, may control specific photons or beams of light. In an example, a processor controls optical strips **155**A-N by initiating mechanical, acoustic, or electromagnetic signals. Although shown as having two optical assemblies **180**A, **180**B, the eyewear device **100** can include other arrangements, such as a single or three optical assemblies, or each optical assembly **180**A, **180**B may have arranged different arrangement depending on the application or intended user of the eyewear device **100**.

As further shown in FIGS. **2**A and **2**B, eyewear device **100** includes a left chunk **110**A adjacent the left lateral side **170**A of the frame **105** and a right chunk **110**B adjacent the right lateral side **170**B of the frame **105**. The chunks **110**A, **110**B may be integrated into the frame **105** on the respective lateral sides **170**A, **170**B (as illustrated) or implemented as separate components attached to the frame **105** on the respective sides **170**A, **170**B. Alternatively, the chunks **110**A, **110**B may be integrated into temples **125**A, **125**B attached to the frame **105**.

In another example, the eyewear device **100** shown in FIG. **2**B may include two projectors, a left projector **150**A (not shown) and a right projector **150**B (shown as projector **150**). The left optical assembly **180**A may include a left display matrix **177**A (not shown) and/or a left set of optical strips **155**'A, **155**'B, . . . **155**'N (**155** prime, A through N, not shown) which are configured to interact with light from the left projector **150**A. Similarly, the right optical assembly **180**B may include a right display matrix **177**B (not shown) and/or a right set of optical strips **155**"A, **155**"B, . . . **155**"N (**155** double-prime, A through N, not shown) which are configured to interact with light from the right projector **150**B. In this example, the eyewear device **100** includes a left display and a right display.

FIG. **4** is a functional block diagram of an example system that includes a wearable device **100** (e.g., an eyewear device), a mobile device **890**, and a server system **998** connected via various networks **995** such as the Internet. The system, as shown, includes a low-power wireless connection **925** and a high-speed wireless connection **937** between the eyewear device **100** and a mobile device **890**.

A computer vision engine **600**, including a mapping module **601**, a color detection module **602**, and a music playback module **603**, are stored in memory **934** for execution by one of the processors **932**, **922** of the eyewear **100**. The eyewear device **100** further includes a user input device **991** (e.g., a touch sensor or touchpad **181**, as shown in FIG. **1**) to receive input from a user.

The eyewear device **100** includes one or more visible-light cameras **114**A, **114**B which may be capable of capturing still images and/or video, as described herein. The cameras **114**A, **114**B may have a direct memory access (DMA) to high-speed circuitry **930**. A pair of cameras **114**A, **114**B may function as a stereo camera, as described herein. The cameras **114**A, **114**B may be used to capture initial-depth images that may be rendered into three-dimensional (3D) models that are texture-mapped images of a red, green, and blue (RGB) imaged scene. The device **100** may also include a depth sensor **213**, which uses infrared signals to estimate the position of objects relative to the device **100**. The depth sensor **213** in some examples includes one or more infrared emitter(s) **215** and infrared camera(s) **220**.

The eyewear device **100** further includes two image displays of each optical assembly **180**A, **180**B (one associated with the left side **170**A and one associated with the right side **170**B). The eyewear device **100** also includes an image display driver **942**, an image processor **912**, low-power circuitry **920**, and high-speed circuitry **930**. The image displays of each optical assembly **180**A, **180**B are for presenting images, including still images and video. The image display driver **942** is coupled to the image displays of each optical assembly **180**A, **180**B in order to control the images displayed.

The components shown in FIG. **4** for the eyewear device **100** are located on one or more circuit boards, for example a PCB or flexible PCB, located in the rims or temples. Alternatively, or additionally, the depicted components can be located in the chunks, frames, hinges, or bridge of the eyewear device **100**. Left and right visible-light cameras **114**A, **114**B can include digital camera elements such as a complementary metal-oxide-semiconductor (CMOS) image sensor, a charge-coupled device, a lens, or any other respective visible or light capturing elements that may be used to capture data, including still images or video of scenes with unknown objects.

As shown in FIG. **4**, high-speed circuitry **930** includes a high-speed processor **932**, a memory **934**, and high-speed wireless circuitry **936**. In the example, the image display driver **942** is coupled to the high-speed circuitry **930** and operated by the high-speed processor **932** in order to drive the left and right image displays of each optical assembly **180**A, **180**B. High-speed processor **932** may be any processor capable of managing high-speed communications and operation of any general computing system needed for eyewear device **100**. High-speed processor **932** includes processing resources needed for managing high-speed data transfers on high-speed wireless connection **937** to a wireless local area network (WLAN) using high-speed wireless circuitry **936**. In certain examples, the high-speed processor **932** executes an operating system such as a LINUX operating system or other such operating system of the eyewear device **100** and the operating system is stored in memory **934** for execution. In addition to any other responsibilities, the high-speed processor **932** executes a software architecture for the eyewear device **100** that is used to manage data transfers with high-speed wireless circuitry **936**. In certain examples, high-speed wireless circuitry **936** is configured to implement Institute of Electrical and Electronic Engineers (IEEE) 802.11 communication standards, also referred to herein as Wi-Fi. In other examples, other high-speed communications standards may be implemented by high-speed wireless circuitry **936**.

The low-power circuitry **920** includes a low-power processor **922** and low-power wireless circuitry **924**. The low-power wireless circuitry **924** and the high-speed wireless circuitry **936** of the eyewear device **100** can include short range transceivers (Bluetooth™) and wireless wide, local, or wide-area network transceivers (e.g., cellular or WiFi). Mobile device **890**, including the transceivers communicating via the low-power wireless connection **925** and the high-speed wireless connection **937**, may be implemented using details of the architecture of the eyewear device **100**, as can other elements of the network **995**.

Memory **934** includes any storage device capable of storing various data and applications, including, among other things, camera data generated by the left and right visible-light cameras **114**A, **114**B, the infrared camera(s) **220**, the image processor **912**, and images generated for display by the image display driver **942** on the image display of each optical assembly **180**A, **180**B. Although the memory **934** is shown as integrated with high-speed circuitry **930**, the memory **934** in other examples may be an independent, standalone element of the eyewear device **100**. In certain such examples, electrical routing lines may provide a connection through a chip that includes the high-speed processor **932** from the image processor **912** or low-power processor **922** to the memory **934**. In other examples, the high-speed processor **932** may manage addressing of memory **934** such that the low-power processor **922** will boot the high-speed processor **932** any time that a read or write operation involving memory **934** is needed.

As shown in FIG. **4**, the high-speed processor **932** of the eyewear device **100** can be coupled to the camera system (visible-light cameras **114**A, **114**B), the image display driver **942**, the user input device **991**, and the memory **934**. As shown in FIG. **5**, the CPU **830** of the mobile device **890** may be coupled to a camera system **870**, a mobile display driver **882**, a user input layer **891**, and a memory **840**A.

The eyewear device **100** can perform all or a subset of any of the functions described herein which result from the execution of the programming (e.g., the computer vision engine **600**, as described herein) which is stored in the memory **934** and executed by one of the processors **932**, **922** of the eyewear device **100**. The mobile device **890** can perform all or a subset of any of the functions described herein which result from the execution of the programming in the flash memory **840**A by the CPU **830** of the mobile device **890**. Functions can be divided such that the mobile device **890** collects motion data using a face-tracking application and sends it to the eyewear device **100**, which performs the rendering functions relative to the display of a virtual object.

The server system **998** may be one or more computing devices as part of a service or network computing system, for example, that include a processor, a memory, and network communication interface to communicate over the network **995** with an eyewear device **100** and a mobile device **890**.

The output components of the eyewear device **100** include visual elements, such as the left and right image displays associated with each lens or optical assembly **180**A, **180**B as described in FIGS. **2**A and **2**B (e.g., a display such as a liquid crystal display (LCD), a plasma display panel (PDP), a light emitting diode (LED) display, a projector, or a waveguide). The eyewear device **100** may include a user-facing indicator (e.g., an LED, a loudspeaker, or a vibrating actuator), and/or an outward-facing signal (e.g., an LED, a loudspeaker). The image displays of each optical assembly **180**A, **180**B are driven by the image display driver **942**. In some example configurations, the output components of the eyewear device **100** further include additional indicators such as audible elements (e.g., loudspeakers), tactile components (e.g., an

actuator such as a vibratory motor to generate haptic feedback), and other signal generators. For example, the device **100** may include a user-facing set of indicators, and an outward-facing set of signals. The user-facing set of indicators are configured to be seen or otherwise sensed by the user of the device **100**. For example, the device **100** may include an LED display positioned so the user can see it, a loudspeaker positioned to generate a sound the user can hear, or an actuator to provide haptic feedback the user can feel. The outward-facing set of signals are configured to be seen or otherwise sensed by an observer near the device **100**. Similarly, the device **100** may include an LED, a loudspeaker, or an actuator that is configured and positioned to be sensed by an observer.

The input components of the eyewear device **100** may include alphanumeric input components (e.g., a touch screen or touchpad configured to receive alphanumeric input, a photo-optical keyboard, or other alphanumeric-configured elements), pointer-based input components (e.g., a mouse, a touchpad, a trackball, a joystick, a motion sensor, or other pointing instruments), tactile input components (e.g., a button switch, a touch screen or touchpad that senses the location and/or force of touches or touch gestures, or other tactile-configured elements), and audio input components (e.g., a microphone), and the like. The mobile device **890** and the server system **998** may include alphanumeric, pointer-based, tactile, audio, and other input components.

In some examples, the eyewear device **100** includes a collection of motion-sensing components referred to as an inertial measurement unit **972**. The motion-sensing components may be micro-electro-mechanical systems (MEMS) with microscopic moving parts, often small enough to be part of a microchip. The inertial measurement unit (IMU) **972** in some example configurations includes an accelerometer, a gyroscope, and a magnetometer. The accelerometer senses the linear acceleration of the device **100** (including the acceleration due to gravity) relative to three orthogonal axes (x, y, z). The gyroscope senses the angular velocity of the device **100** about three axes of rotation (pitch, roll, yaw). Together, the accelerometer and gyroscope can provide position, orientation, and motion data about the device relative to six axes (x, y, z, pitch, roll, yaw). The magnetometer, if present, senses the heading of the device **100** relative to magnetic north. The position of the device **100** may be determined by location sensors, such as a GPS receiver, one or more transceivers to generate relative position coordinates, altitude sensors or barometers, and other orientation sensors. Such positioning system coordinates can also be received over the wireless connections **925**, **937** from the mobile device **890** via the low-power wireless circuitry **924** or the high-speed wireless circuitry **936**.

The IMU **972** may include or cooperate with a digital motion processor or programming that gathers the raw data from the components and compute a number of useful values about the position, orientation, and motion of the device **100**. For example, the acceleration data gathered from the accelerometer can be integrated to obtain the velocity relative to each axis (x, y, z); and integrated again to obtain the position of the device **100** (in linear coordinates, x, y, and z). The angular velocity data from the gyroscope can be integrated to obtain the position of the device **100** (in spherical coordinates). The programming for computing these useful values may be stored in memory **934** and executed by the high-speed processor **932** of the eyewear device **100**.

The eyewear device **100** may optionally include additional peripheral sensors, such as biometric sensors, specialty sensors, or display elements integrated with eyewear device **100**. For example, peripheral device elements may include any I/O components including output components, motion components, position components, or any other such elements described herein. For example, the biometric sensors may include components to detect expressions (e.g., hand expressions, facial expressions, vocal expressions, body gestures, or eye tracking), to measure biosignals (e.g., blood pressure, heart rate, body temperature, perspiration, or brain waves), or to identify a person (e.g., identification based on voice, retina, facial characteristics, fingerprints, or electrical biosignals such as electroencephalogram data), and the like.

The mobile device **890** may be a smartphone, tablet, laptop computer, access point, or any other such device capable of connecting with eyewear device **100** using both a low-power wireless connection **925** and a high-speed wireless connection **937**. Mobile device **890** is connected to server system **998** and network **995**. The network **995** may include any combination of wired and wireless connections.

The system, as shown in FIG. **4**, includes a computing device, such as mobile device **890**, coupled to an eyewear device **100** and to a mobile device **890** over a network. Execution of the programming instructions by the processor **932** configures the eyewear device **100** to cooperate with the mobile device **890**. The system may utilize the memory **934** of the eyewear device **100** and/or the memory elements **840**A, **840**B of the mobile device **890** (FIG. **5**). Also, the system may utilize the processor elements **932**, **922** of the eyewear device **100** and/or the central processing unit (CPU) **830** of the mobile device **890** (FIG. **5**). Furthermore, the system may further utilize the memory and processor elements of the server system **998**. In this aspect, the memory and processing functions can be shared or distributed across the eyewear device **100**, the mobile device **890**, and/or the server system **998**.

FIG. **5** is a high-level functional block diagram of an example mobile device **890**. Mobile device **890** includes a flash memory **840**A which stores programming to be executed by the CPU **830** to perform all or a subset of the functions described herein. As shown, a computer vision engine **600** may reside on the CPU **830** and accessed for operating one or more applications or modules, one or more of which may be stored in the memory **840**A. The mobile device **890** may include an IMU **872** and a loudspeaker **895** for playing sounds.

The mobile device **890** may include a camera **870** that comprises at least two visible-light cameras (first and second visible-light cameras with overlapping fields of view) or at least one visible-light camera and a depth sensor with substantially overlapping fields of view. Flash memory **840**A may further include multiple images or video, which are generated via the camera **870**.

As shown, the mobile device **890** includes an image display **880**, a mobile display driver **882** to control the image display **880**, and a controller **894**. In the example of FIG. **4**, the image display **880** includes a user input layer **891** (e.g., a touchscreen) that is layered on top of or otherwise integrated into the screen used by the image display **880**.

Examples of touchscreen-type mobile devices that may be used include (but are not limited to) a smart phone, a personal digital assistant (PDA), a tablet computer, a laptop computer, or other portable device. However, the structure and operation of the touchscreen-type devices is provided by way of example; the subject technology as described herein is not intended to be limited thereto. For purposes of this discussion, FIG. **4** therefore provides a block diagram illus-

tration of the example mobile device **890** with a user interface that includes a touchscreen input layer **891** for receiving input (by touch, multi-touch, or gesture, and the like, by hand, stylus or other tool) and an image display **880** for displaying content.

As shown in FIG. **4**, the mobile device **890** includes at least one digital transceiver (XCVR) **810**, shown as WWAN XCVRs, for digital wireless communications via a wide-area wireless mobile communication network. The mobile device **890** also includes additional digital or analog transceivers, such as short range XCVRs **820** for short-range network communication, such as via NFC, VLC, DECT, ZigBee, Bluetooth™, or WiFi. For example, short range XCVRs **820** may take the form of any available two-way wireless local area network (WLAN) transceiver of a type that is compatible with one or more standard protocols of communication implemented in wireless local area networks, such as one of the Wi-Fi standards under IEEE 802.11.

To generate location coordinates for positioning of the mobile device **890**, the mobile device **890** can include a global positioning system (GPS) receiver. Alternatively, or additionally the mobile device **890** can utilize either or both the short range XCVRs **820** and WWAN XCVRs **810** for generating location coordinates for positioning. For example, cellular network, Wi-Fi, or Bluetooth™ based positioning systems can generate very accurate location coordinates, particularly when used in combination. Such location coordinates can be transmitted to the eyewear device over one or more network connections via XCVRs **810**, **820**.

The transceivers **810**, **820** (i.e., the network communication interface) conforms to one or more of the various digital wireless communication standards utilized by modern mobile networks. Examples of WWAN transceivers **810** include (but are not limited to) transceivers configured to operate in accordance with Code Division Multiple Access (CDMA) and 3rd Generation Partnership Project (3GPP) network technologies including, for example and without limitation, 3GPP type 2 (or 3GPP2) and LTE, at times referred to as "4G." For example, the transceivers **810**, **820** provide two-way wireless communication of information including digitized audio signals, still image and video signals, web page information for display as well as web-related inputs, and various types of mobile message communications to/from the mobile device **890**.

The mobile device **890** further includes a microprocessor that functions as a central processing unit (CPU); shown as CPU **830** in FIG. **4**. A processor is a circuit having elements structured and arranged to perform one or more processing functions, typically various data processing functions. Although discrete logic components could be used, the examples utilize components forming a programmable CPU. A microprocessor for example includes one or more integrated circuit (IC) chips incorporating the electronic elements to perform the functions of the CPU. The CPU **830**, for example, may be based on any known or available microprocessor architecture, such as a Reduced Instruction Set Computing (RISC) using an ARM architecture, as commonly used today in mobile devices and other portable electronic devices. Of course, other arrangements of processor circuitry may be used to form the CPU **830** or processor hardware in smartphone, laptop computer, and tablet.

The CPU **830** serves as a programmable host controller for the mobile device **890** by configuring the mobile device **890** to perform various operations, for example, in accordance with instructions or programming executable by CPU **830**. For example, such operations may include various general operations of the mobile device, as well as operations related to the programming for applications on the mobile device. Although a processor may be configured by use of hardwired logic, typical processors in mobile devices are general processing circuits configured by execution of programming.

The mobile device **890** includes a memory or storage system, for storing programming and data. In the example, the memory system may include a flash memory **840**A, a random-access memory (RAM) **840**B, and other memory components, as needed. The RAM **840**B serves as short-term storage for instructions and data being handled by the CPU **830**, e.g., as a working data processing memory. The flash memory **840**A typically provides longer-term storage.

Hence, in the example of mobile device **890**, the flash memory **840**A is used to store programming or instructions for execution by the CPU **830**. Depending on the type of device, the mobile device **890** stores and runs a mobile operating system through which specific applications are executed. Examples of mobile operating systems include Google Android, Apple iOS (for iPhone or iPad devices), Windows Mobile, Amazon Fire OS, RIM BlackBerry OS, or the like.

Computer Vision and Mapping

FIG. **6** is an illustration of an example map **700**, including a marker **730**, a note matrix **710**, a percussion matrix **770**, and a tempo scale **750**, suitable for use with the computer vision engine **600** described herein. The map **700** according to some implementations is part of a method of playing a musical composition.

The marker **730** is an icon or other indicia which provides a known starting point when an instrument such as a camera **870** captures a color image of the map **700**. The marker **730** establishes a map coordinate system for all the elements on the map **700**. For example, the location of the marker **730** can be used to locate the position of the note matrix **710**, and other elements, relative to the map coordinate system. The size and scale of the map coordinate system may vary depending on the size of the map **700**; for example, when the map **700** is printed onto a sheet of paper or another surface. In one example implementation, the act of capturing the marker **730** with a camera prompts the electronic device to download the computer vision engine **600**.

The note matrix **710**, as shown, includes a plurality of rows and columns. Each row is associated with a note; most often, a note in a musical scale. For example, a fourth row **712**A is associated with a fourth note **714**A (the note, F, in this example). Notes are different from sounds. Any of a variety of musical instruments can play a particular note, although each instrument will have a unique sound.

Each column in the note matrix **710** is associated with a beat duration. In music, a beat is the basic unit of time. A series of beats creates a rhythm, which is most often a steady rhythm made up of beats having the same duration. The speed of the beats is known as the tempo. The beat duration for the first column **716**A, for example, may be one second.

The rows and columns of the note matrix **710** define a plurality of fields **720**. For example, the note matrix **710** in FIG. **6** includes eight rows and eight columns, defining sixty-four fields. Each field in the note matrix **710** is characterized by a set of field coordinates relative to the map coordinate system (which is based on the marker **730**). When a map **700** is printed on a sheet of paper and laid flat, or onto another flat surface, the map coordinate system may be two-dimensional.

The computer vision engine **600** in some implementations includes a mapping module **601** which may accomplish the steps of recognizing the marker **730**, setting the map coordinate system, and then identifying the set of field coordinates associated with each field **720** in the note matrix **710**.

A color image captured by a digital camera is used to detect the positions, colors, and other attributes of each field **720** and other elements on the map **700**. The color image may be shown on the display screen of a mobile device **890**, as described herein. The location of specific points in the color image, therefore, may be described using a two-dimensional display coordinate system that is associated with the display screen of the mobile device **890**. The mapping module **601** in some implementations converts the set of field coordinates for each field to a pair of display coordinates. In this aspect, the coordinates for each field on paper are converted into coordinates for each field on the display.

Tokens **760** are placed on the note matrix **710** in order to create a musical composition. FIG. **7** is an illustration of a set of example tokens **760** which are ready to be cut and placed on a note matrix **710**. In the example, a first token **760**A is associated with a first sound sample for a keyboard instrument, such as a plano. A second token **760**B is associated with a second sound sample for a brass instrument, such as a cornet. A third token **760**C is associated with a third sound sample for a string instrument, such as a viola. Although illustrated as squares, the tokens **760** may be any size and shape suitable for a particular map **700**. The tokens **760** are illustrated as shaded squares in FIG. **7**; however, instead of shading, each type of token **760** has a particular token color. For example, the first token **760**A may be red in color, such that a token color of red is associated with the first sound sample for a keyboard instrument.

The computer vision engine **600** in some implementations includes a color detection module **602** which may accomplish the step of detecting a token color value for each field **720** in the note matrix **710**. The color detection module **602** analyzes the color image of the map **700** that was captured by the camera. The color image captured by a digital camera includes raw data about the color at each location or pixel. Each color may be expressed as a set of RGB values, also known as an RGB triplet. A color in the RGB color model is described by indicating how much of each of red (R), green (G), and blue (B) color is present. For a digital image, 8 bits per channel, the amount of each color is expressed as an integer between zero and 255. The color white, for example, is expressed as RGB (255, 255, 255); a mixture of all three colors. The color red is expressed as RGB (255, 0, 0). The numerical RGB values are useful when comparing colors based on their relative saturation and intensity.

The token color value for each field **720** determines the musical instrument sound to be played, the row where a token **760** is placed determines the note **712** to be played, and the column **716** determines the beat or time at which the sound is to be played.

For example, in the note matrix **710** shown in FIG. **6**, a first token **760**A is placed in the fourth row **712**A and in the first column **716**A. The first token **760**A, as shown, is not perfectly centered in the field; in other words, the token may not be perfectly aligned with the pair of display coordinates for the field. The color detection module **602** detects the token color value that is near the display coordinates. According to the example note matrix **710**, for the first beat of music (associated with the first column **716**A), the presence of the first token **760**A is detected in the fourth row **712**A (associated with the fourth note **714**A (note F)), and the first token color value of the first token **760**A is detected as red (RGB 255, 0, 0; which is associated with the sounds of a keyboard instrument). During playback, as described herein, a sound sample for a keyboard instrument plays the note F on the first beat; and so on, for each beat (column) in sequence until completed.

In use, any number of tokens **760** of various colors may be placed anywhere on the note matrix **710**. To record the position and color of each token, the computer vision engine **600** stores in memory a global state map, which includes the token color value (as detected by the color detection module **602**) and the pair of display coordinates for each field in the note matrix **710**. In this aspect, the global state map acts as a guide during playback (by the music playback module **603**).

Each example token **760**, as shown in FIG. **7**, is associated with a sound sample associated with a particular musical instrument. The computer vision engine **600** in some implementations includes the step of storing in the memory a plurality of preset token color values, where each one is associated with a preset sound sample. The values may be stored in a table or database, for example. The first token **760**A may be associated with a first preset token color value of pure red (RGB 255, 0, 0), with a first sound sample from a keyboard instrument. In use, the first sound sample will include, of course, the sound of each note in a musical scale (because a first token **760**A may be placed in any row **712** of the note matrix **710**).

The sound samples for each instrument, including a variety of notes, may be obtained from any of a variety of available sound libraries. The computer vision engine **600** in some examples may be configured to record user-created sounds, such as the user's singing voice, other vocal sounds, and percussive sounds such as hand claps and finger snaps.

Printing: The token color value of a token **760** printed in color on a paper usually does not have the same RGB value as the preset token color value. For example, the first preset token color value of pure red (RGB 255, 0, 0) when printed on paper may have different RGB values. Depending on the printer, ink colors, paper type, and a host of other variables, the actual token color value of a first token **760**A printed on paper may not be pure red. The printed first token **760**A may appear red on paper, but in the digital color image it may include other colors: for example, the red may be (RGB 231, 111, 96). The color detection module **602** in some implementations performs the step of calculating a difference between the token color value on paper (RGB 231, 111, 96) and one or more of the plurality of preset token color values (which includes pure red, RGB 255, 0, 0). The calculated difference is used to select the preset token color value that is closest to the token color value as it appears on paper.

The background color value also varies depending on the printer and it is usually not the same as pure white (RGB 255, 255, 255). The color detection module **602** in some implementations performs the step of detecting the background color value associated with an unmarked area **739** of the map **700** as printed in color on a sheet of paper or other surface. The unmarked area in some implementations is part of the marker **730**. The color detection module **620** then calculates a ratio between the background color value on paper and the preset background color value (typically, pure white RGB 255, 255, 255).

For many printers, this ratio can be used to correct or otherwise adjust some or all of the other color values on the map **700**. The color detection module **620** multiplies each token color value by the ratio to obtain a modified token color value. Using the modified values, the color detection

module **620** calculates a variance between each modified token color value and one or more of the plurality of preset token color values. The variance is used to select the preset token color value that is closest to the token color value as it appears on paper.

Referring again to FIG. **4**, the map **700** may also include a percussion matrix **770**. The columns of the percussion matrix **770** are aligned with the columns of the note matrix **710**, as shown. Each rhythm row **772** is associated with a percussion sound. For example, a second row **772**B is associated with a hand-clap sound. The rhythm rows **772** do not refer to notes because percussion sounds are not associated with notes on a scale. The rows and columns define a plurality of rhythm fields. The mapping module **601** as described herein identifies the rhythm field coordinates associated with each rhythm field.

The color detection module **620** in some implementations detects whether a percussion token **778** is present or not at each rhythm field location. In this aspect, the rhythm attribute value for each rhythm field is a binary variable; either one or zero. The percussion token **778** may be a particular color or no color. According to the example percussion matrix **770** in FIG. **6**, for the first beat of music (associated with the first column **716**A), the presence of the percussion token **778** is detected in the second rhythm row **772**B (associated with a hand-clap sound). During playback, as described herein, a percussion sound sample for a hand clap is played on the first beat—at the same time a sound sample for a keyboard instrument plays the note F on the first beat (based on the note matrix **710**). In this aspect, the playing of percussion sounds is substantially correlated in time with said playing of the sound samples for instruments.

Referring again to FIG. **4**, the map **700** may also include a pointer **752** located along a tempo scale **750** that extends lengthwise from a minimum speed to a maximum speed. The mapping module **601** as described herein may be used to identify the pointer coordinates relative to the tempo scale **750**. The music playback module **603** may adjust the beat duration (associated with each column **716**) according to the pointer coordinates, such that the relative location of the pointer **752** determines the playback tempo.

In some implementations, the music playback module **603** may be set to play the music in each column **716**, in sequence, through and including the last column. The playing may be repeated in a continuing loop. The music playback module **603** uses the global state map to determine the selection and sequence of notes and sounds to be played. The global state map in some implementations is persistent; that is, it remains set unless and until the map **700** is changed. In this aspect, the mobile device **890** may be directed away from the map **700**, so that the map **700** is not within the field of view of the camera **870**; nevertheless, the stored global state map will persist and the playback will continue, as stored.

When changes are made to the map **700**, the computer vision engine **600** is configured to periodically capture a subsequent color image (of the new map), store in the memory an updated global state map, and then play the music in accordance with the updated global state map.

In a related aspect, the computer vision engine **600** may include a hand detection algorithm that will pause the step of storing an updated global state map while and for as long as the algorithm detects a hand in any subsequent color image. In other words, if and when a hand is detected in a subsequent color image (of the new map), the computer vision engine **600** will not attempt to detect a color value for the hand or otherwise build an updated global state map.

In some implementations, the map **700** is placed in the field of view of a digital camera **870** of a mobile device **890**. Accordingly, the color image of the map **700** is shown (continuously, in the usual case) on the display screen of the mobile device **890**. The mobile device **890** may be mounted on a tripod or otherwise held in position relative to the map **700** during use. The music playback module **603** in some implementations is configured to project a virtual object on the display screen according to the global state map. The virtual object may include any of a variety of graphical elements, colors, or other indicia. For example, the virtual object may be a glowing square or other highlight that appears near the field of the note or sound when it is played.

In a related aspect, the map **700** may be located in the field of view of one or more lenses **180**A, **180**B of a wearable device, such as an eyewear device **100**. The eyewear device **100** may include a projector and a semi-transparent display. The music playback module **603** in some implementations is configured to project a virtual object onto the semi-transparent display, using the global state map and the coordinates associated with the semi-transparent display to place the virtual object in a desired location. In this aspect, the computer vision engine **600** is configured to overlay the virtual object relative to a physical environment, as part of an augmented reality.

A wearable device, such as eyewear **100**, may include a number of loudspeakers or other assemblies that provide spatial audio capability. In use, particular sounds may seem to originate from one or more locations in space relative to the wearer. Because the position of the map **700** relative to the eyewear **100** is known, the sounds of the instruments during playback may be spatialized.

Any of the functionality described herein for the eyewear device **100**, the mobile device **890**, and the server system **998** can be embodied in one or more computer software applications or sets of programming instructions, as described herein. According to some examples, "function," "functions," "application," "applications," "instruction," "instructions," or "programming" are program(s) that execute functions defined in the programs. Various programming languages can be employed to create one or more of the applications, structured in a variety of manners, such as object-oriented programming languages (e.g., Objective-C, Java, or C++) or procedural programming languages (e.g., C or assembly language). In a specific example, a third-party application (e.g., an application developed using the ANDROID™ or IOS™ software development kit (SDK) by an entity other than the vendor of the particular platform) may include mobile software running on a mobile operating system such as IOS™, ANDROID™, WINDOWS® Phone, or other mobile operating systems. In this example, the third-party application can invoke API calls provided by the operating system to facilitate functionality described herein.

Hence, a machine-readable medium may take many forms of tangible storage medium. Non-volatile storage media include, for example, optical or magnetic disks, such as any of the storage devices in any computer devices or the like, such as may be used to implement the client device, media gateway, transcoder, etc. shown in the drawings. Volatile storage media include dynamic memory, such as main memory of such a computer platform. Tangible transmission media include coaxial cables; copper wire and fiber optics, including the wires that comprise a bus within a computer system. Carrier-wave transmission media may take the form of electric or electromagnetic signals, or acoustic or light waves such as those generated during radio frequency (RF) and infrared (IR) data communications. Common forms of

computer-readable media therefore include for example: a floppy disk, a flexible disk, hard disk, magnetic tape, any other magnetic medium, a CD-ROM, DVD or DVD-ROM, any other optical medium, punch cards paper tape, any other physical storage medium with patterns of holes, a RAM, a PROM and EPROM, a FLASH-EPROM, any other memory chip or cartridge, a carrier wave transporting data or instructions, cables or links transporting such a carrier wave, or any other medium from which a computer may read programming code and/or data. Many of these forms of computer readable media may be involved in carrying one or more sequences of one or more instructions to a processor for execution.

Except as stated immediately above, nothing that has been stated or illustrated is intended or should be interpreted to cause a dedication of any component, step, feature, object, benefit, advantage, or equivalent to the public, regardless of whether it is or is not recited in the claims.

It will be understood that the terms and expressions used herein have the ordinary meaning as is accorded to such terms and expressions with respect to their corresponding respective areas of inquiry and study except where specific meanings have otherwise been set forth herein. Relational terms such as first and second and the like may be used solely to distinguish one entity or action from another without necessarily requiring or implying any actual such relationship or order between such entities or actions. The terms "comprises," "comprising," "includes," "including," or any other variation thereof, are intended to cover a non-exclusive inclusion, such that a process, method, article, or apparatus that comprises or includes a list of elements or steps does not include only those elements or steps but may include other elements or steps not expressly listed or inherent to such process, method, article, or apparatus. An element preceded by "a" or "an" does not, without further constraints, preclude the existence of additional identical elements in the process, method, article, or apparatus that comprises the element.

Unless otherwise stated, any and all measurements, values, ratings, positions, magnitudes, sizes, and other specifications that are set forth in this specification, including in the claims that follow, are approximate, not exact. Such amounts are intended to have a reasonable range that is consistent with the functions to which they relate and with what is customary in the art to which they pertain. For example, unless expressly stated otherwise, a parameter value or the like may vary by as much as plus-or-minus ten percent relative to the stated value.

In addition, in the foregoing Detailed Description, it can be seen that various features are grouped together in various examples for the purpose of streamlining the disclosure. This method of disclosure is not to be interpreted as reflecting an intention that the claimed examples require more features than are expressly recited in each claim. Rather, as the following claims reflect, the subject matter to be protected lies in less than all features of any single disclosed example. Thus, the following claims are hereby incorporated into the Detailed Description, with each claim standing on its own as a separately claimed subject matter.

While the foregoing has described what are considered to be the best mode and other examples, it is understood that various modifications may be made therein and that the subject matter disclosed herein may be implemented in various forms and examples, and that they may be applied in numerous applications, only some of which have been described herein. It is intended by the following claims to claim any and all modifications and variations that fall within the true scope of the present concepts.

What is claimed is:

1. A method of playing a musical composition, said method comprising:

initiating a computer vision and mapping application on a mobile device, said mobile device comprising a processor, a memory, a camera, a loudspeaker, and a display screen characterized by a two-dimensional display coordinate system;

storing in said memory a plurality of preset token color values and sets of sound samples associated with musical instruments, wherein a first preset token color value is associated with a first set of sound samples and a first musical instrument;

capturing with said camera a color image of a marker and a note matrix on a map printed on a surface, wherein said note matrix comprises a plurality of rows, each row associated with a note in a musical scale, and a plurality of columns, each column associated with a beat duration, such that said note matrix defines a plurality of fields;

recognizing said marker in said color image and establishing a map coordinate system relative to said marker;

identifying a set of field coordinates associated with each of said plurality of fields of said note matrix relative to said map coordinate system;

converting said set of field coordinates to a set of display coordinates associated with each of said plurality of fields relative to said two-dimensional display coordinate system;

detecting in said color image a plurality of token color values associated with each of said plurality of fields;

storing in said memory a global state map comprising said plurality of detected token color values and said set of display coordinates associated with each of said plurality of fields;

determining whether said plurality of detected token color values includes said first preset token color value;

in response to determining that said plurality of detected token color values includes said first preset token color value, retrieving from said memory said first set of sound samples associated with said first musical instrument;

determining, for each of said plurality of columns, whether a first token is positioned near a select column and near a select row, wherein said select row is associated with a select note in a musical composition;

in response to determining that said first token is positioned near said select row, retrieving from said memory a select sound sample from said retrieved first set of sound samples; and

initiating a music playback module on said mobile device, wherein said music playback module is configured to play said musical composition according to each of said plurality of columns in said note matrix, in a sequence from left to right, comprising playing through said loudspeaker said select sound sample associated with said select column.

2. The method of claim **1**,

wherein said step of determining whether said plurality of detected token color values includes said first preset token color value comprises:

calculating a difference between a first token color value associated with said first token printed in color on a paper and said first preset token color value; and

determining that said plurality of detected token color values includes said first preset token color value if said difference is less than a threshold value.

3. The method of claim 1, wherein said step of determining whether said plurality of detected token color values includes said first preset token color value further comprises:

detecting a background color value associated with an unmarked area on said map;

calculating a ratio between said background color value and a preset background color value;

multiplying said plurality of detected token color values by said ratio to obtain a modified set of detected token color values;

calculating a variance between said modified set of detected token color values and said first preset token color value; and

determining whether said plurality of detected token color values includes said first preset token color value based on said variance.

4. The method of claim 1, wherein said map further comprises a percussion matrix comprising said plurality of columns of said note matrix together with one or more rhythm rows, each rhythm row associated with a percussion sound, said percussion matrix defining a plurality of rhythm fields,

said method further comprising:

storing in said memory a preset rhythm attribute value and a set of percussion sound samples associated with a percussion instrument, wherein a first preset rhythm attribute is associated with a first set of percussion sound samples and a first percussion instrument;

identifying rhythm field coordinates associated with each of said plurality of rhythm fields;

detecting a plurality of rhythm attribute values associated with each of said plurality of rhythm fields, wherein each of said detected rhythm attribute values comprises one or zero;

updating said global state map in said memory such that said global state map further comprises said plurality of detected rhythm attribute values and said rhythm field coordinates associated with each of said plurality of rhythm fields; and

determining whether said plurality of detected token rhythm attribute values includes said first preset rhythm attribute;

in response to determining that said plurality of detected token rhythm attribute values includes said first preset rhythm attribute, retrieving from said memory said first set of percussion sound samples associated with said first percussion instrument;

determining, for each of said plurality of columns, whether a first rhythm token is positioned near said select column and near a select rhythm row, wherein said select rhythm row is associated with a select percussion sound in a musical composition;

in response to determining that said first rhythm token is positioned near said select rhythm row, retrieving from said memory a select percussion sound sample from said retrieved first set of percussion sound samples; and

wherein said music playback module is further configured to play said musical composition according to each of said plurality of columns in said note matrix, in sequence from left to right and for said beat duration, comprising playing through said loudspeaker said select percussion sound sample associated with said select column.

5. The method of claim 1, wherein said map further comprises a pointer located along a tempo scale extending lengthwise from a minimum speed to a maximum speed,

wherein said step of playing in a sequence further comprises:

identifying pointer coordinates associated with said pointer relative to said tempo scale; and

adjusting said beat duration according to said pointer coordinates.

6. The method of claim 1, wherein said step of capturing with said camera a color image further comprises:

periodically capturing a subsequent color image;

storing in said memory an updated global state map according to said subsequent color image; and

executing said step of playing in accordance with said updated global state map.

7. The method of claim 6, further comprising:

initiating a hand detection algorithm on said mobile device; and

pausing said step of storing an updated global state map while and for as long as said algorithm detects a hand in said subsequent color image.

8. The method of claim 1, further comprising:

displaying said color image on said display screen of said mobile device; and

for each of said plurality of columns in said note matrix, in a sequence from left to right and for said beat duration, projecting a virtual object on said display screen according to said global state map.

9. The method of claim 8, further comprising:

viewing said map through one or more lenses of a wearable device, said wearable device comprising a projector and a semi-transparent display characterized by a wearable coordinate system;

converting said pair of display coordinates associated with each of said plurality of fields relative to said two-dimensional display coordinate system to a set of coordinates relative to said wearable coordinate system; and

projecting said virtual object onto said semi-transparent display according to said set of relative coordinates.

10. The method of claim 1, further comprising:

determining whether said plurality of detected token color values includes a subsequent preset token color value;

in response to determining that said plurality of detected token color values includes said subsequent preset token color value, retrieving from said memory a subsequent set of sound samples associated with a subsequent musical instrument;

determining, for each of said plurality of columns, whether a subsequent token is positioned near a subsequent column and near a subsequent row, wherein said subsequent row is associated with a subsequent note in said musical composition;

in response to determining that said subsequent token is position near said subsequent row, retrieving from said memory a subsequent sound sample from said retrieved subsequent set of sound samples; and

wherein said step of playing through said loudspeaker further comprises playing through said loudspeaker said subsequent sound sample associated with said subsequent column.

11. A system comprising:

a mobile device comprising a processor, a memory, a camera, a loudspeaker, and a display screen characterized by a two-dimensional display coordinate system;

programming in said memory, wherein execution of said programming by said processor configures said mobile device to perform functions, including functions to:

store in said memory a plurality of preset token color values and sets of sound samples associated with musical instruments, wherein a first preset token color value is associated with a first set of sound samples and a first musical instrument;

capture with said camera a color image of a marker and a note matrix on a map printed on a surface, wherein said note matrix comprises a plurality of rows, each row associated with a note in a musical scale, and a plurality of columns, each column associated with a beat duration, such that said note matrix defines a plurality of fields;

recognize said marker in said color image and establishing a map coordinate system relative to said marker;

identify a set of field coordinates associated with each of said plurality of fields of said note matrix relative to said map coordinate system;

convert said set of field coordinates to a set of display coordinates associated with each of said plurality of fields relative to said two-dimensional display coordinate system;

detect in said color image a plurality of token color values associated with each of said plurality of fields;

store in said memory a global state map comprising said plurality of detected token color values and said set of display coordinates associated with each of said plurality of fields;

determine whether said plurality of detected token color values includes said first preset token color value;

in response to determining that said plurality of detected token color values includes said first preset token color value, retrieve from said memory said first set of sound samples associated with said first musical instrument;

determine, for each of said plurality of columns, whether a first token is positioned near a select column and near a select row, wherein said select row is associated with a select note in a musical composition;

in response to determining that said first token is positioned near said select row, retrieve from said memory a select sound sample from said retrieved first set of sound samples; and

initiate a music playback module on said mobile device, wherein said music playback module is configured to play said musical composition according to each of said plurality of columns in said note matrix, in a sequence from left to right, comprising playing through said loudspeaker said select sound sample associated with said select column.

12. The system of claim **11**, wherein said execution further configures said mobile device to:

calculate a difference between a first token color value associated with said first token printed in color on a paper and said first preset token color value; and

determine that said plurality of detected token color values includes said first preset token color value if said difference is less than a threshold value.

13. The system of claim **11**, wherein said execution further configures said mobile device to:

detect a background color value associated with an unmarked area on said map;

calculate a ratio between said background color value and a preset background color value;

multiply said plurality of detected token color values by said ratio to obtain a modified set of detected token color values;

calculate a variance between said modified set of detected token color values and said first preset token color value; and

determine whether said plurality of detected token color values includes said first preset token color value based on said variance.

14. The system of claim **11**, wherein said map further comprises a percussion matrix comprising said plurality of columns of said note matrix together with one or more rhythm rows, each rhythm row associated with a percussion sound, said percussion matrix defining a plurality of rhythm fields, and wherein said execution further configures said mobile device to:

store in said memory a preset rhythm attribute value and a set of percussion sound samples associated with a percussion instrument, wherein a first preset rhythm attribute is associated with a first set of percussion sound samples and a first percussion instrument;

identify rhythm field coordinates associated with each of said plurality of rhythm fields;

detect a plurality of rhythm attribute values associated with each of said plurality of rhythm fields, wherein each of said detected rhythm attribute values comprises one or zero;

update said global state map in said memory such that said global state map further comprises said plurality of detected rhythm attribute values and said rhythm field coordinates associated with each of said plurality of rhythm fields;

determine whether said plurality of detected token rhythm attribute values includes said first preset rhythm attribute;

in response to determining that said plurality of detected token rhythm attribute values includes said first preset rhythm attribute, retrieve from said memory said first set of percussion sound samples associated with said first percussion instrument;

determine, for each of said plurality of columns, whether a first rhythm token is positioned near said select column and near a select rhythm row, wherein said select rhythm row is associated with a select percussion sound in a musical composition;

in response to determining that said first rhythm token is positioned near said select rhythm row, retrieve from said memory a select percussion sound sample from said retrieved first set of percussion sound samples; and

play said musical composition according to each of said plurality of columns in said note matrix, in sequence from left to right and for said beat duration, la playing through said loudspeaker said select percussion sound sample associated with said select column.

15. The system of claim **11**, wherein said map further comprises a pointer located along a tempo scale extending lengthwise from a minimum speed to a maximum speed, and wherein said execution further configures said mobile device to:

identify pointer coordinates associated with said pointer relative to said tempo scale; and

adjust said beat duration according to said pointer coordinates.

16. The system of claim **11**, wherein said execution further configures said mobile device to:

periodically capture a subsequent color image;

store in said memory an updated global state map according to said subsequent color image; and

execute said step of playing in accordance with said updated global state map.

17. The system of claim **16**, wherein said execution further configures said mobile device to:

initiate a hand detection algorithm on said mobile device; and

pause said step of storing an updated global state map while and for as long as said algorithm detects a hand in said subsequent color image.

18. The system of claim **11**, wherein said execution further configures said mobile device to:

display said color image on said display screen of said mobile device; and

for each of said plurality of columns in said note matrix, in a sequence from left to right and for said beat duration, project a virtual object on said display screen according to said global state map.

19. The system of claim **18**, wherein said execution further configures said mobile device to:

view said map through one or more lenses of a wearable device, said wearable device comprising a projector and a semi-transparent display characterized by a wearable coordinate system;

convert said pair of display coordinates associated with each of said plurality of fields relative to said two-dimensional display coordinate system to a set of coordinates relative to said wearable coordinate system; and

project said virtual object onto said semi-transparent display according to said set of relative coordinates.

20. The system of claim **11**, wherein said execution further configures said mobile device to:

determine whether said plurality of detected token color values includes a subsequent preset token color value;

in response to determining that said plurality of detected token color values includes said subsequent preset token color value, retrieve from said memory a subsequent set of sound samples associated with a subsequent musical instrument;

determine, for each of said plurality of columns, whether a subsequent token is positioned near a subsequent column and near a subsequent row, wherein said subsequent row is associated with a subsequent note in said musical composition;

in response to determining that said subsequent token is position near said subsequent row, retrieve from said memory a subsequent sound sample from said retrieved subsequent set of sound samples; and

play through said loudspeaker said subsequent sound sample associated with said subsequent column.

21. A non-transitory computer-readable medium storing program code which, when executed, is operative to cause an electronic processor to perform the steps of:

storing in said memory a plurality of preset token color values and sets of sound samples associated with musical instruments, wherein a first preset token color value is associated with a first set of sound samples and a first musical instrument;

capturing with a camera of a mobile device a color image of a marker and a note matrix on a map printed on a surface, wherein said note matrix comprises a plurality of rows, each row associated with a note in a musical scale, and a plurality of columns, each column associated with a beat duration, such that said note matrix defines a plurality of fields;

recognizing said marker in said color image and establishing a map coordinate system relative to said marker;

identifying a set of field coordinates associated with each of said plurality of fields of said note matrix relative to said map coordinate system;

converting said set of field coordinates to a set of display coordinates associated with each of said plurality of fields relative to said two-dimensional display coordinate system;

detecting in said color image a plurality of token color values associated with each of said plurality of fields;

storing in a memory a global state map comprising said plurality of detected token color values and said set of display coordinates associated with each of said plurality of fields;

determining whether said plurality of detected token color values includes said first preset token color value;

in response to determining that said plurality of detected token color values includes said first preset token color value, retrieving from said memory said first set of sound samples associated with said first musical instrument;

determining, for each of said plurality of columns, whether a first token is positioned near a select column and near a select row, wherein said select row is associated with a select note in a musical composition;

in response to determining that said first token is positioned near said select row, retrieving from said memory a select sound sample from said retrieved first set of sound samples; and

playing said musical composition according to each of said plurality of columns in said note matrix, in a sequence from left to right, comprising playing through said loudspeaker said select sound sample associated with said select column.

22. The non-transitory computer-readable medium storing program code of claim **21**, wherein said execution of the program code is operative to further cause the electronic processor to perform the steps of:

wherein said step of determining whether said plurality of detected token color values includes said first preset token color value comprises:

calculating a difference between a first token color value associated with said first token printed in color on a paper and said first preset token color value; and

determining that said plurality of detected token color values includes said first preset token color value if said difference is less than a threshold value.

* * * * *